Ricardo Filipe Ferreira Soares

U. PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO
DEPARTAMENTO DE ENGENHARIA E GESTÃO INDUSTRIAL

New Models and Methods for the Vehicle Routing Problem with Multiple Synchronisation Constraints

Submitted to Faculdade de Engenharia da Universidade do Porto in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering and Industrial Management, supervised by Alexandra Marques, Senior Researcher at INESC TEC, and Pedro Amorim, Associate Professor of Faculdade de Engenharia da Universidade do Porto

Department of Industrial Engineering and Management
Faculdade de Engenharia da Universidade do Porto
2022

This research was partially supported by the PhD Grant SFRH/BD/136314/2018,
awarded by Fundação para a Ciência e a Tecnologia (FCT),
the Portuguese Foundation for Science and Technology.

"Plans are worthless,
but planning is everything."

- Dwight D. Eisenhower

Acknowledgments

First and foremost, I would like to thank my supervisors. This process would not have been the same without your vote of confidence and long-term commitment to guiding me through this multiple-year journey. Alexandra provided me with the opportunity to enter the research world, encouraging me to view my PhD as a process of personal and professional growth. Her unique perspective on issues strengthened my critical thinking and helped me to keep my eyes on the prize. Pedro, with his constant optimism, always pushed me one step further when I thought I was at a roadblock. Whenever I met with him, I knew I would never leave the meeting the same way I entered. I am deeply grateful to both of you.

I would also like to thank Prof. Sophie Parragh. This PhD would also be very different without all the discussions and insights she provided me. Although circumstances did not allow us to collaborate in the way we wanted, future opportunities will surely come up.

The team at INESC TEC was also very important in reaching this milestone. I thank my team colleagues Carolina Catorze and Reinaldo Gomes for their invaluable support, patience, and friendly work atmosphere throughout these years. I also need to thank Maria João Santos and Sara Martins for their friendliness and gratuitous assistance in overcoming many of the hurdles I faced in this journey.

Finally, I direct the last acknowledgement to my family, particularly my parents. Your values, guidance and unconditional support throughout the years have made me the person that was able to get here. You are the bravest people I know and inspire me to be a better person every day.

Abstract

Across different value chains, transportation and logistics are at the core of research agendas, motivated by their impact on economic, social and environmental terms. In particular, the Vehicle Routing Problem (VRP) has been an effervescent field of research for the past 60 years. This can be attributed not only to the $\mathcal{NP}$-hard combinatorial nature of the problem, thus constituting a challenge when solving it, but also to its practical relevance to organisations when planning their logistics and transportation activities. Despite the obtained breakthroughs by the Operations Research (OR) community, businesses still face increasingly specific and complex transportation requirements that need to be tackled. In this context, one of the challenges that have been emerging is synchronisation.

The VRP with Synchronisation is a generalisation of the VRP where the feasibility of a route is dependent on the feasibility of another route. Examples of these problems typically include requirements where different vehicles must temporally synchronise their arrival to a customer (e.g., scheduling patient homecare) – either at the same time or within given time intervals – or when different vehicles need to be travelling simultaneously – as it happens with problems using tractors and trailers, for example.

This thesis advances the state of the art on the topic of vehicle routing with synchronisation. Initially, this thesis systematises knowledge on the topic. This research was grounded on qualitative analyses based on an integrative literature review, which is also used to identify relevant research gaps.

This thesis also tackles specific and practical challenges identified in the literature. One of these challenges consists of modelling and solving complex routing problems with multiple types of synchronisation constraints. To that effect, a systematic approach is proposed, motivated by a real-life case-study. The obtained results from this research provide valuable managerial insights that can be implemented in the context of the case-study.

Furthermore, this thesis also tackles VRPs with Synchronisation acknowledging sources of uncertainty. Based on consolidated knowledge from the literature, modelling and solution approaches are developed and tested in a set of computational experiments.

This thesis is complemented by a supplementary material section that presents relevant research work performed by the PhD candidate in the course of his training.

Resumo

Em diversas cadeias de valor, as atividades de transporte e logística encontram-se no topo das prioridades de investigação, motivadas pelos seus impactos a nível económico, social e ambiental. Em particular, o Problema de Roteamento de Veículos (VRP) tem sido alvo de investigação intensa nos últimos 60 anos. Este facto pode ser justificado não apenas à natureza combinatória $\mathcal{NP}$-hard deste problema, constituindo assim um desafio para o resolver, mas também à sua relevância prática no planeamento das atividades logísticas e de transporte das organizações. Apesar dos avanços obtidos pela comunidade de Investigação Operacional (IO), as empresas ainda enfrentam requisitos de transporte cada vez mais específicos e complexos que precisam de ser atendidos. Um desses desafios que têm vindo a emergir é a sincronização de veículos.

O VRP com Sincronização é uma generalização do VRP onde a admissibilidade de uma rota depende da admissibilidade de outra rota. Exemplos desses problemas geralmente incluem restrições operacionais em que diferentes veículos devem sincronizar temporalmente a sua chegada a um cliente (e.g., marcação de assistência de serviços de saúde ao domicílio) – ao mesmo tempo ou dentro de determinados intervalos de tempo – ou quando diferentes veículos precisam de circular simultaneamente – como acontece com problemas de roteamento que recorrem a tratores e reboques, por exemplo.

Esta tese avança o estado da arte no tópico de roteamento de veículos com sincronização. Inicialmente, esta tese sistematiza o conhecimento sobre este tema de investigação. Este trabalho foi baseado em análises qualitativas baseadas numa revisão integrativa da literatura, que é também utilizada para identificar lacunas de investigação relevantes.

Esta tese também aborda desafios específicos e práticos identificados na literatura. Um desses desafios consiste em modelar e resolver problemas complexos de roteamento com vários tipos de restrições de sincronização. Propõe-se uma abordagem sistemática, motivada por um caso de estudo real.

Os resultados obtidos nesta pesquisa permitem obter conclusões valiosas que podem ser implementadas no âmbito do caso de estudo. Além disso, esta tese também aborda VRPs com Sincronização com fontes de incerteza. Com base no conhecimento consolidado da literatura, são desenvolvidas abordagens de modelação e métodos de solução, sendo os mesmos testados num conjunto de testes computacionais.

Esta tese é ainda complementada por um conjunto de apêndices que apresenta trabalhos de investigação relevantes que foram realizados pelo doutorando no decurso da sua formação.

Contents

Chapter 1

Motivation and overview

For the past decade, modern societies have felt the effects of globalisation in their daily lives, to which technological advancements had an indisputable contribution. In order to maintain its relevance in markets with progressively more competitors from various geographies, organisations have acknowledged the importance of having efficient and effective practices in managing their supply chain activities.

One of the major activities present in supply chains is transportation, as it is essential for linking its different echelons. Due to its ubiquity in traditional supply chains, transportation activities can constitute a significant part of the total logistics costs [Kherbach and Mocan, 2016].

The activities of transportation planning are intimately linked with planning vehicle routes. The Vehicle Routing Problem (VRP) is a highly-known combinatorial optimisation problem whose purpose is to determine the optimal set of routes for a given set of vehicles so that a given set of customers is visited exactly once with minimum transportation costs. The Operations Research (OR) community has extensively studied VRPs for the past 60 years, ever since the seminal paper of Dantzig and Ramser [1959] and several variants were developed throughout the years to accommodate different operational realities.

One of the VRP topics that has been gaining increasing momentum is the VRP with synchronisation aspects. It consists of a routing problem where a given set of routes is dependent on each other for the problem to become feasible. This problem usually translates into the need for different vehicles to meet in locations at the same time or within given time offsets or even travelling portions of a route simultaneously. The route interdependence present in the VRP with synchronisation is not present in standard VRPs and constitutes a difficulty when developing solution methods for these types of problems, as a change in one vehicle's route may render all other routes infeasible. However, synchronisation in VRPs can be viewed as a very advantageous requirement to consider in transportation plans. The main advantage of synchronisation consists in the reduction of waiting/queueing times for routes that are interdependent, thus increasing asset utilisation and potentiating the increase of customer service level. In this case, VRPs with synchronisation can enable the exploration of new solution spaces and contribute to more cost-effective routing plans. Ultimately, the reduction of unproductive times can be such that organisations may be able to reduce the overall vehicle fleet size needed to perform its transportation operations.

Drexl [2012] provided the first comprehensive survey concerning VRPs with synchronisation and proposed a classification schema for different types of synchronisation. The presented framework allowed to systematise different types of use cases for inter-vehicle synchronisation, such as requiring two or more vehicles to arrive at a location in a given order or within a given time offset, or ensuring that a given non-autonomous vehicle (e.g., trailer) is transported by another vehicle for it to be moved between locations.

These routing problems often appear in the context of real-life applications and tend to be covered in application-specific literature. Ioachim et al. [1999] tackle an aircraft fleet scheduling problem, where it is necessary to guarantee that similar flights have equal departure times throughout the week. More recent publications tackle this synchronisation aspect, such as Bredström and Rönnqvist [2008] and Mankowska et al. [2013], addressing a home health care routing and scheduling problem (HHCRSP) where multiple staff members are required to arrive at a patient's home at the same time or within a given time offset in order to perform the requested health care services. Additionally, simultaneous movement of vehicles (i.e., ensuring that non-autonomous vehicles are transported by other vehicles) is typically found within the VRP variant of the Truck and Trailer Routing Problem (TTRP), where the vehicle fleet consists of truck units and trailer units with some customers only accessible by truck. For that purpose, trailers can be uncoupled *en route* at customers where truck sub-tours are built (e.g., Chao [2002], Parragh and Cordeau [2017]).

The topic of VRPs with synchronisation is still underdeveloped. Despite the systematisation effort of Drexl [2012], synchronisation is still a broad concept in the literature, being frequently used for problem aspects that belong to any standard VRP.

One of the research questions of this thesis clarifies the concept of synchronisation by taking the definition presented by Drexl [2012] and clarifying its underlying differences to other problem aspects found in VRPs. Furthermore, the fact that there is no common modelling framework for VRPs with synchronisation difficults the generalisation of these problems to a problem variant that is independent of the application. This research gap is also addressed in this thesis by providing a unified modelling framework for VRPs with synchronisation, which can be applicable to several problem variants in the literature.

This thesis also performs research and advances the state of the art on new real-life applications of VRPs with synchronisation. In fact, there are relevant real-life settings where more than one type of synchronisation exists. This doctoral research also performs breakthroughs on this front by modelling and solving a real-life case-study in the biomass supply chain, involving different types of vehicles and several synchronisation requirements between them.

Finally, the research stream of VRPs with synchronisation also lacks studies on approaches to handle uncertain data and sources of stochasticity. This is an especially relevant topic for this type of routing problem, as route interdependence can make any slight deviation from the original plan propagate infeasibilities to the entire routing plan. This doctoral research also tackles this research gap by constructing a methodology for addressing uncertainty in routing problems with synchronisation.

1.1. Research objectives

This research will focus on different challenges that the VRP with synchronisation aspects entails. To that effect, a total of three research questions will be answered, which are represented in Figure 1.1 and described ahead.

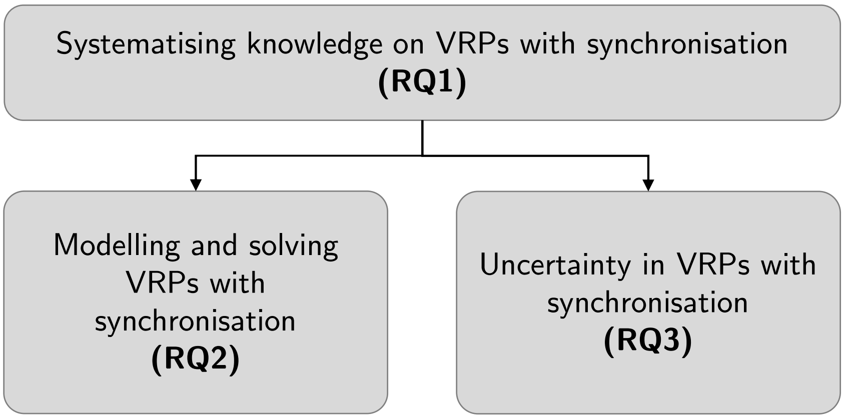


Figure 1.1: Research canvas

The first research question of this thesis is:

Research question 1 (RQ1)
How to systematise knowledge on synchronisation aspects within the vehicle routing problem?

The topic of synchronisation in vehicle routing was firstly approached by Drexl [2012]. The author conducted a survey for a new class of VRPs, called VRP with Multiple Synchronisation Constraints (VRPMS), and proposed a new classification framework for different synchronisation aspects. The survey helped to streamline several nomenclatures for synchronisation (which were non-existent at the time) across the literature. Despite this effort, the literature still uses synchronisation in a broad sense, covering a wide range of problems. Furthermore, VRPs with synchronisation are typically application-driven and tend to be tailor-made for the problem itself, being hard to replicate to different cases.
To bridge these deficiencies in the literature, this research question will refocus the concept of synchronisation in VRPs, building a classification schema for different VRP problem aspects, including synchronisation, which considers the core definition proposed by Drexl [2012]. This research question also envisages the development of a systematic approach to model VRPs with multiple synchronisation aspects, which will be built from the proposed classification schema.

The second research question of this thesis is:

Research question 2 (RQ2)
How can we model and solve routing problems with multiple synchronisation constraints?

This research question is motivated by the fact that the vast majority of routing problems with synchronisation only acknowledges one type of synchronisation aspects. This research question intends to provide a systematic approach to model and solve routing problems with multiple synchronisation aspects, involving multiple types of vehicles performing different tasks, based on a real-life case-study.

The third and final research question of this thesis is:

Research question 3 (RQ3)
How can we address uncertainty in routing problems with synchronisation?

Applications where some information is subject to uncertainty can have a profound impact on the deployment of a routing problem with synchronisation. The literature is scarce regarding approaches for these types of problems. This research question intends to bridge this gap by developing a methodology to solve and address stochastic information in routing problems with synchronisation.

1.2. Research methodology

The adopted research methodologies for the conduction of this research depended on the specific research questions being addressed.
For answering Research Question 1, a strong work of conceptualisation was required. The conceptualisation work of this task was achieved through a literature review of the existing applications of synchronisation in routing problems. To that effect, searches for articles were performed in scientific databases. The search queries for articles were carefully tuned to avoid getting large amounts of literature that is irrelevant for the subject at hand. Afterwards, the formal aspects of each paper were analysed, such as scope, journal type and date of publication. For assessing the scope of each publication, a preliminary analysis of the abstract was instrumental in deciding whether the publication should be considered or not. Each paper was afterwards carefully read to have a final assessment of its scope. Finally, the results of this literature review were summarised into a conceptual schema, whose criteria were clearly defined, and each publication was framed within that same conceptual schema.
Throughout all research questions, problem modelling was an activity of extreme relevance. This was be accomplished mainly through the development of mixed integer programming (MIP) formulations, eventually considering different objective functions and constraints, depending on the insights the project intends to obtain. Existing mathematical formulations from the literature were adopted and abridged to the specific needs of each research question.
Research questions 2 and 3 also encompassed a strong effort not only on problem modelling but also on the development of solution methods. Regarding the development of solution methods, the overall approach consisted in using both exact and approximate methods. The former was typically used for testing MIP models and obtaining optimal solutions

for small-scale instances. The latter approach was used when addressing real-world problems, as exact methods are not suitable for large-size problems of combinatorial nature. The development of approximate methods focused on the conception of heuristic solution methods hybridised with mathematical solvers. The different models and solution methods developed were tested, when possible, for real-based instances. The developed models and methods were assessed and compared with respect to computational efficiency and solution quality. To that effect, indicators traditionally used in the literature were used, such as solution time and average solution gap.

1.3. Thesis structure and synopsis

This thesis is structured as a collection of research papers and it is composed of five chapters and two appendices. This section provides an overview of each of the chapters and appendices that comprise this thesis.

Chapter 1 provides a concise overview of the research topic, stating the research's motivation and relevance. The objectives of the research are also stated along with the research questions being addressed. The overall methodology on which the research was performed is also explained. Finally, the overall structure of the thesis is outlined, summarising the contents and findings of each chapter.

Chapter 2 is entitled "Systematising knowledge on routing problems with synchronisation" and addresses Research Question 1. Since the inception of the VRP, synchronisation in vehicle routing has been present in specific real-life problems and applications throughout the years. However, these problems had not been consolidated into a general class of routing problems, nor had they been classified as synchronisation problems until the survey of Drexl [2012], which established the class of VRPs with Multiple Synchronisation Constraints. This survey established the interdependence factor as the critical requirement for a routing problem to be considered a synchronisation problem. However, despite this systematisation effort, the concept of synchronisation still got used broadly. The scientific contributions of this chapter are three-fold and are grounded on an integrative literature review. First, building on the concepts of Drexl [2012], a clear definition of synchronisation within vehicle routing is provided. This theoretical definition is instrumental toward a narrower and less ambiguous understanding of the synchronisation concept within the Operations Research (OR) community. The research builds on previously existing classification schemas, leveraging their concepts to develop a comprehensive classification schema for different VRP problem aspects, clearly separating synchronisation aspects. Second, this chapter updates the state of the art on routing problems with synchronisation, classifies the types of synchronisation found in the literature, provides an overview of the adopted solution approaches and identifies potential research gaps, opportunities and trends. Finally, to further contribute to a general class of VRPs synchronisation, this chapter presents a unified modelling framework for routing problems with synchronisation aspects. The reference that serves as a basis for this chapter is:

- Ricardo Soares, Alexandra Marques, Pedro Amorim, and Sophie N. Parragh. Synchronisation in vehicle routing: a literature-based classification schema and modelling framework. *European Journal of Operational Research* (Submitted)

Chapter 3 is entitled "Modelling and solving routing problems with synchronisation" and directly addresses Research Question 2. This chapter provides original research on an innovative application of OR and tackles a routing problem with multiple synchronisation requirements involving multiple types of vehicles. The problem is motivated by a real-life application in the forest sector, specifically in biomass transportation planning, which is designated as the biomass "hot-system". The biomass "hot-system" assumption requires that wood chipping operations, usually performed by the forest roadside, be executed without any intermediate storage so that chipped residues are directly loaded onto the container of a truck. This assumption requires that chipping machines and trucks be temporally synchronised. In addition, the chipping machines must be transported between forest roadside locations, as they cannot move autonomously, thus requiring a set of lorries to transport them. The problem is tackled through a compact formulation of a mixed integer programming model. Model complexity is minimised by resorting to custom transportation network generation procedures and pre-processing routines. Afterwards, larger problem instances are solved by resorting to a fix-and-optimise matheuristic, thus combining features from both exact and approximate methods. Finally, managerial insights are drawn from computational experiments designed to analyse different possible scenarios. The reference that serves as a basis for this chapter is:

- Ricardo Soares, Alexandra Marques, Pedro Amorim, and Jussi Rasinmäki. Multiple vehicle synchronisation in a full truck-load pickup and delivery problem: A case-study in the biomass supply chain. *European Journal of Operational Research*, 277(1):174–194, 2019.

Chapter 4 is entitled "Dealing with uncertainty in routing problems with synchronisation" and directly addresses Research Question 3. Most approaches tackling vehicle routing problems are of a deterministic nature. However, depending on specific problem applications and instances, disregarding relevant sources of uncertainty from planning processes may turn out to be costly and reduce the relevancy of a routing plan. Considering the inter-route dependency relationship that exists in routing problems with synchronisation, it is possible that a random variation in one of the problem's parameters may propagate to multiple routes. This chapter investigates a robust optimisation approach to tackle the VRP subject to synchronisation, specifically operations synchronisation, considering uncertain travel times. Robust optimisation is an approach that accounts for uncertain parameters when their probability distribution functions are unknown. Instead, bounded uncertainty sets are considered, where the worst-case values are typically assumed to obtain a "robust-feasible" solution. However, considering worst-case values for all uncertain parameters is highly conservative and irrealistic for most problem settings. To avoid this situation, a budget of uncertainty is typically considered, which represents the minimum number of worst-case occurrences that a robust-feasible solution must be able to hold, regardless of where they occur in the problem. The budget of uncertainty is, therefore, a measure of the degree of conservativeness that a robust-feasible solution must hold. This chapter builds on an existing approach in the literature for considering robustness in the problem. The modelling framework developed in Chapter 2 of this thesis is leveraged and adapted to present a mixed integer programming model for the Robust VRP with synchronisation. To solve this problem, a branch-and-cut algorithm is presented. The solution approach considers multiple specificities of the routing problem to generate relevant cuts that are injected into the mathematical solver. The reference that serves as a basis for this chapter is:

- Ricardo Soares, Sophie N. Parragh, Pedro Amorim, and Alexandra Marques. A robust optimisation approach for the vehicle routing problem with synchronisation. *Working paper*, 2022.

Chapter 5 provides a summary of the main conclusions and contributions of this thesis, highlighting the fulfilment of the research questions and stating future research opportunities and paths.

This thesis also includes two additional chapters containing supplementary material. These two appendices consist of research papers where the PhD candidate is a co-author, having had a significant scientific contribution to them. Although these complementary papers are not directly linked with the scope of the research topic of this thesis, they have contributed to the PhD candidate's training within the scope of the doctoral programme.
Appendix A is entitled "Network design for transportation planning with synchronisation". This chapter addresses a tactical problem for the biomass supply chain where it is necessary to determine the optimal flows of biomass residues from biomass sources – piles of biomass by the forest roadside – to its consumers – bioenergy centrals. Besides the assignment of biomass residues, decisions related to the assignment of chipping machines to forest sites are also considered. A mixed integer programming model is proposed to solve this problem to optimality. A set of computational experiments is devised to test the performance of a state-of-the-art mathematical solver. Sensitivity analyses were also devised to compare solution results and obtain valuable managerial insights for the biomass sector. The case-study analysed in this paper is intimately linked with the one in chapter 3 of this thesis. In this research paper, the scientific contributions provided by the PhD candidate concerned the development and implementation of the mathematical models. The reference that serves as a basis for this chapter is:

- Alexandra Marques, Jussi Rasinmäki, Ricardo Soares, and Pedro Amorim. Planning woody biomass supply in hot systems under variable chips energy content. *Biomass and Bioenergy*, 108:265–277, 2017.

Appendix B is entitled "Integrating inbound and outbound transportation planning". This chapter addresses a routing problem whose purpose is to evaluate the potential benefits of integrating the outbound and inbound logistics of a wood-based panel industry. The problem is tackled as a VRP with Backhauls with rich problem requirements specific to the case-study, namely specific rules for visiting backhaul customers, split deliveries to customers and the use of a heterogeneous fleet. Different outbound and inbound integration strategies were tested, such as the complete integration of outbound and inbound flows or the opportunistic integration of inbound flows with the outbound flows. A set of computational experiments was envisaged to test and compare each of these strategies with the baseline scenario, where no integration exists. For solving larger problem instances, a fix-and-optimise solution approach was envisaged. The scientific contributions of the PhD candidate in this research paper concerned the design of the inbound and outbound strategies being studied, the development and implementation of the mathematical models and the development of the matheuristic solution approach. The reference that serves as a basis for this chapter is:

- Alexandra Marques, Ricardo Soares, Maria João Santos, and Pedro Amorim. Integrated planning of inbound and outbound logistics with a Rich Vehicle Routing Problem with backhauls. *Omega*, 92, 102172, 2019.

Bibliography

David Bredström and Mikael Rönnqvist. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, 191(1):19–31, 2008. doi: 10.1016/j.ejor.2007.07.033.

I-Ming Chao. A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29(1):33–51, 2002. doi: 10.1016/s0305-0548(00)00056-3.

G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6 (1):80–91, 1959. doi: 10.1287/mnsc.6.1.80.

Michael Drexl. Synchronization in vehicle routing—a survey of VRPs with multiple synchronization constraints. *Transportation Science*, 46(3):297–316, 2012. doi: 10.1287/trsc.1110.0400.

Irina Ioachim, Jacques Desrosiers, François Soumis, and Nicolas Bélanger. Fleet assignment and routing with schedule synchronization constraints. *European Journal of Operational Research*, 119(1):75–90, 1999. doi: 10.1016/s0377-2217(98)00343-9.

Oualid Kherbach and Marian Liviu Mocan. The importance of logistics and supply chain management in the enhancement of romanian SMEs. *Procedia - Social and Behavioral Sciences*, 221:405–413, 2016. doi: 10.1016/j.sbspro.2016.05.130.

Dorota Slawa Mankowska, Frank Meisel, and Christian Bierwirth. The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, 17(1):15–30, 2013. doi: 10.1007/s10729-013-9243-1.

Sophie N. Parragh and Jean-François Cordeau. Branch-and-price and adaptive large neighborhood search for the truck and trailer routing problem with time windows. *Computers & Operations Research*, 83:28–44, 2017. doi: 10.1016/j.cor.2017.01.020.

CHAPTER 2

Systematising knowledge on routing problems with synchronisation

---

Synchronisation in vehicle routing: a literature-based classification schema and modelling framework

Ricardo Soares [*†] · **Alexandra Marques**[*] · **Pedro Amorim**[*†] · **Sophie N. Parragh**[‡]

Submitted to *European Journal of Operational Research*, 2021.

Abstract The practical relevance and challenging nature of the Vehicle Routing Problem (VRP) have motivated the Operations Research community to consider different practical requirements and problem variants throughout the years. However, businesses still face increasingly specific and complex transportation requirements that need to be tackled, one of them being synchronisation. No literature contextualises synchronisation among other types of problem aspects of the VRP, increasing ambiguity in the nomenclature used by the community. The contributions of this paper originate from a literature review and are threefold. First, new conceptual and classification schemas are envisaged to extend previous work in the literature and re-organise different interdependencies that arise in routing decisions. To that effect, different problem aspects in the VRP literature are characterised, situating and clarifying the concepts of synchronisation by outlining different types of interdependencies that arise in VRPs. Secondly, to enable the development of a general class of VRPs with synchronisation, a modelling framework is presented based on the previously proposed schemas. Finally, the literature review results are leveraged to identify future research gaps and opportunities in the field of VRPs with synchronisation.

Keywords routing · synchronisation · literature review · classification schema · mathematical programming

[*]INESC TEC – Institute for Systems and Computer Engineering, Technology and Science, Porto, Portugal
[†]FEUP – Faculty of Engineering, University of Porto, Porto, Portugal
[‡]Institute of Production and Logistics Management / JKU Business School, Johannes Kepler University Linz, Linz, Austria

2.1. Introduction

Businesses face increasingly fierce competition where lead times and flexibility are crucial factors for customer satisfaction. In this context, transportation processes need to account for more and more specific and complex requirements, which may be achieved by maintaining transportation processes *as-is* and investing in additional resources (e.g., inventory, vehicles) to buffer the impacts. However, this approach can be economically unsustainable for some business sectors, thus triggering the need for alternative approaches that "squeeze" the most out of the available resources. Furthermore, new practical settings for routing problems have been emerging throughout the years where the specialisation of resources, vehicles or staff becomes a key and unavoidable factor in planning processes [e.g., Paraskevopoulos et al., 2017].

Synchronisation of vehicles constitutes a clear and practical opportunity for these operational realities. It enables the coordination of vehicles for specific routing tasks that are dependent on other vehicles. In complex supply chains, where multiple vehicles, crews, materials and other resources are involved, synchronisation can be a catalyst for more efficiently combining different operations that require different resources by decreasing unproductive times. This can eventually result in a decrease in the overall resources required. The topic of synchronisation in vehicle routing was first approached systematically by Drexl [2012], proposing a classification schema of synchronisation aspects, where interdependencies arise in the routing decisions concerning two or more different vehicles. The survey characterised the VRP with Multiple Synchronisation Constraints (VRPMS) as a VRP where routes were dependent on one another, as opposed to standard VRPs. Consequently, a change in one route may affect other routes, which, in a worst-case scenario, may render all other routes infeasible. The classification schema of Drexl [2012] was built upon a specific case study of the VRP with Trailers and Transhipments (VRPTT). Through induction, five main types of synchronisation were proposed: task, operation, movement, load and resource synchronisation.

Despite the systematisation provided by Drexl [2012], the concept of synchronisation is still used in a broad manner [e.g., Gschwind, 2015, Bolduc et al., 2006, Russell et al., 2008], beyond the scope of interdependence between different routes, as proposed by Drexl [2012]. There is a lack of consensus on the different types of synchronisation one can find in a VRP. Moreover, there are even three of the categories proposed by Drexl [2012] – task, load and resource synchronisation – that are common to the general class of VRPs and not at all specific to cases where two dependent routes are considered.

The ambiguity behind the concept of synchronisation in VRPs constitutes a major obstacle to consolidating knowledge in this field. To that effect, the purpose of this paper consists in answering the following research questions:

1. How can one define synchronisation in the Vehicle Routing Problem and distinguish it from other problem aspects?

2. What are the upcoming trends and research opportunities in the field of VRPs that contain synchronisation aspects?

3. How can one use the definition of synchronisation to provide a general class of Vehicle Routing Problems with synchronisation aspects?

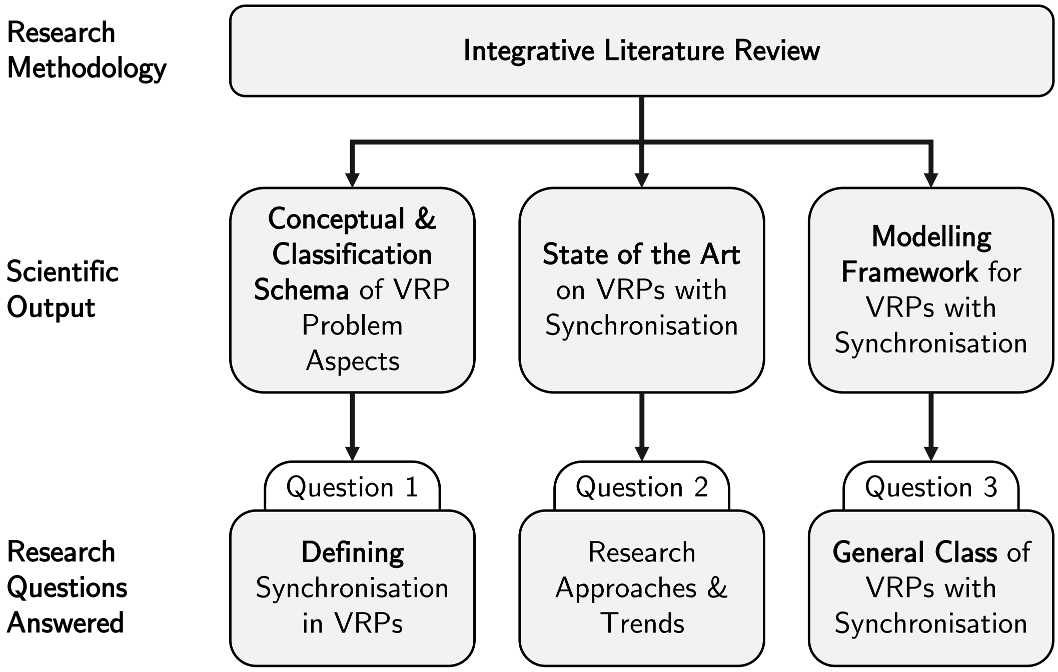


Figure 2.1: Visual overview of the utilised research methodology and scientific results

The link between the research questions and the scientific contribution of this article is visually organised in Figure 2.1. As illustrated in the figure, all of the scientific output of this paper is grounded on a literature review. The produced review consists of one that can be classified as an *integrative literature review* [Snyder, 2019]. Its purpose is to summarise and organise the literature on the topic of VRPs with synchronisation to allow the development of new perspectives and theoretical frameworks. The survey performed by Drexl [2012] provided a stepping stone toward this goal, to which the authors intend to further contribute.

The first research question of this paper intends to re-focus the concept of synchronisation on what derives from the definition from Drexl [2012]: routes are interdependent, i.e., vehicles are not independent of each other – consequently, changes in a route may affect the feasibility of other routes. To that effect, we build on the concepts of Drexl [2012] and formulate new conceptual and classification schemas to clearly define boundaries between what should be considered synchronisation and what should not, depending on the interdependencies that arise for given problem aspects. The resulting schemas are general and can be applied to most VRPs in the literature. The classification schema, in particular, constitutes a comprehensive way to classify different VRP aspects, with the advantage of clearly separating synchronisation aspects. This clear separation of synchronisation aspects from other problem aspects is the main distinctive factor of our classification schema, which, to the best of our knowledge, is not present in any other classification schemas in the literature. Nevertheless, the proposed schemas build on already existing perspectives in the literature, such as the classification schema of Irnich et al. [2014] for common VRP aspects such as "local" or "global" constraints, as well as the categorisation of synchronisation aspects provided by Drexl [2012].

The second research question intends to update the state of the art of this research stream. Using the previously proposed conceptual and classification schemas and the literature review, relevant publications considering routing problems with synchronisation are organised, taking into account their similarities. Furthermore, we provide an overview of adopted solution approaches for these problems, among other relevant aspects and research gaps. From the performed literature review, opportunities and trends are identified, which may be explored by the research community in the future.

Finally, the third and last research gap this paper intends to address consists in the development of a general class of VRPs with synchronisation aspects. Probably due to the difficulties in modelling and solving such types of routing problems, the modelling approaches seen in the literature are usually tailor-made to the application being studied because the focus of such approaches is on the scalability and efficiency to obtain good quality solutions. However, this makes it difficult to generalise these problems to a common, well-defined class within the VRP literature. Although the review by Drexl [2012] references some additional modelling endeavours that these problems need to acknowledge, this work can be complemented with a modelling framework that can clarify and streamline these synchronisation aspects in terms of its modelling efforts. This can ultimately serve as a starting point for modelling new routing problems with synchronisation and potentially using it in solution approaches that resort to mathematical programming. With that in mind, we leverage this paper's conceptual and classification schemas and convey it to a modelling framework for the VRP with synchronisation that can be used in most real-world applications of this problem variant. We present a highly modular mathematical formulation that mingles synchronisation aspects with other common problem aspects in the VRP literature, such as demand satisfaction, capacity constraints, and time windows.

The remainder of this paper is as follows. Section 2.2 presents the outline of the literature review by describing the adopted methodology and a holistic overview of its results. Section 2.3 presents conceptual and classification schemas for classifying different problem aspects in VRPs, focusing on the classification of synchronisation aspects. Section 2.4 presents the main results of the literature review on the topic of VRPs with synchronisation. Section 2.5 proposes a unified modelling framework for modelling VRPs with synchronisation, based on the previously proposed conceptual schema. Section 2.6 instantiates the modelling framework into well-known VRP variants of the literature. Finally, Section 2.7 provides a discussion of the results obtained in this paper and suggests possible research avenues for future work.

2.2. Literature review methodology

The conduction of the literature review followed an approach similar to the one in Seuring et al. [2005], comprising four main stages: material collection, descriptive analysis, category selection and material evaluation. Despite these well-defined stages, the overall process was highly iterative, given the research questions of this paper and the integrative nature of the review. The scope of the literature review concerned VRPs with at least one synchronisation aspect.

Material collection. The collection of the references that served as a basis for the literature review was performed from two sources of information. The first source of information was the Scopus database in January 2022 with the following search query: `(“vehicle routing” OR “arc routing” OR “pickup and delivery” OR “truck and trailer”) AND (“synchronization” OR “temporal dependencies” OR “temporal interdependencies”)`. The selection of these keywords was performed iteratively by testing several combinations of keywords commonly used in the VRP field and assessing the overall relevance of the obtained references. Additionally to the results obtained from Scopus, we added all the references that cited Drexl [2012], as this was the first seminal paper that approached synchronisation aspects in the VRP in a systematic manner. Additional criteria for the search included filtering the results by only including publications written in the English language. From these criteria, a total of 294 references were initially retrieved, which were later reduced to a shortlist of 100 references in subsequent stages of the review, based on the criteria described below.

Descriptive analysis. In order to evaluate the relevance and interest of the topic, the metadata of the references were analysed. The focus of this analysis was mainly on the year of publication and the types of journals where the research was published.
Figure 2.2 shows the distribution of the papers that were reviewed by year of publication. As observable in this figure, the increasing trend in the number of papers throughout the years supports the claim that synchronisation is a growing research stream.
Figure 2.3 exhibits the distribution of papers throughout the journals. Although two clearly stand out, one can observe a great dispersion of the papers throughout several journals. The most commonly found journals are *Computers and Operations Research* and the *European Journal of Operational Research*. These journals support the hypothesis that the research community is aware that additional work on tackling the topic of synchronisation in VRPs is necessary, as these journals have a high focus on methodological contributions for decision making.

Category selection. The category selection phase was instrumental in designing the classification schema proposed in this paper. This stage aimed to devise a categorisation of each problem aspect one may find in a VRP. It was established that the most appropriate way to clarify synchronisation aspects would be to take a holistic categorisation for most VRP problem aspects, such that it can be used to categorise both VRPs with and without synchronisation.
The categorisation was initially built on Irnich et al. [2014], who consider VRP aspects either as “local” (or “intra-route”) constraints or “global” (or “inter-route”) constraints. Local constraints relate to problem aspects that can be checked for their feasibility once an individual route in a VRP is known, regardless of what happens in other routes. In other words, these are the aspects that establish whether or not a given route is feasible. On

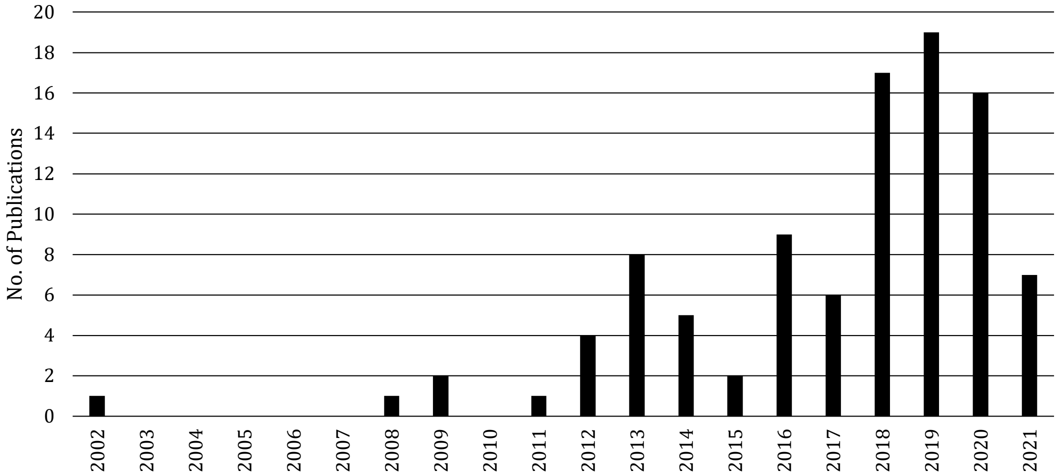


Figure 2.2: Distribution of the reviewed papers by year of publication

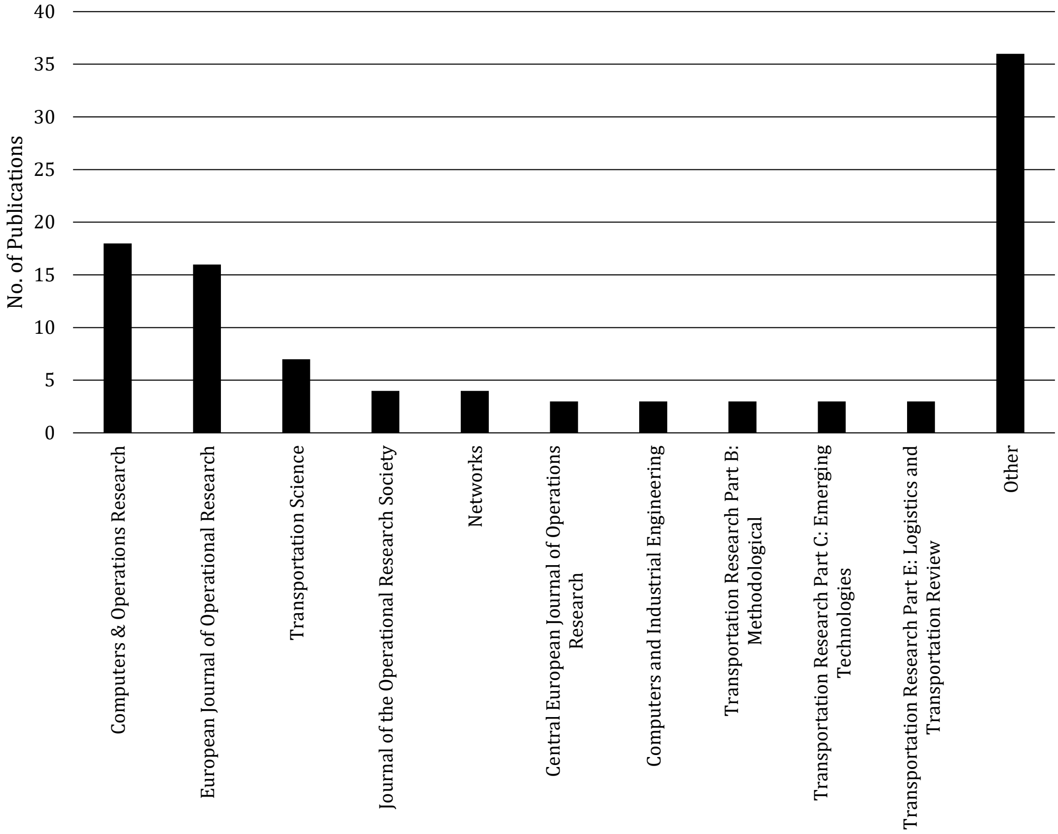


Figure 2.3: Distribution of the reviewed papers by journal

the other hand, global constraints relate to problem aspects whose feasibility depends on the global design of the routes present in a solution. In other words, these are the aspects where specific resources are globally available and therefore need to be rationed by vehicles/routes.
As the material evaluation stage progressed, the categories were iteratively refined. This led to an iterative readjustment of the concept of synchronisation in vehicle routing, with repercussions both on the selected papers and their classification. As similar problem aspects were found throughout the literature, a set of general VRP concepts was also devised, which serve as a basis for the classification schema and allow to homogenise nomenclature. The final conceptual and classification schemas are presented in Section 2.3.

Material evaluation. Each of these references was manually filtered through an analysis of the abstracts and the type of publication. Criteria for exclusion of a publication could be one of the following: (i) the publication did not showcase a routing problem with synchronisation aspects, according to our classification schema; (ii) the publication was not a research paper containing original research (e.g., the publication was a survey/review); (iii) the publication consisted of ongoing or unfinished research work (e.g., conference proceedings), and it was not published recently (i.e., from 2020 onwards); (iv) the publication lacked a formal definition of the problem being tackled, either through a mathematical model or a detailed problem statement section (or equivalent). The authors reached a shortlist of 100 references using these filtering criteria, which were then categorised according to the classification schema. Additional information was taken regarding the application of the problem being tackled, the objective function being optimised, and the adopted solution approach, if proposed. This information was cleansed and presented in Appendix 2.B.

2.3. Conceptual and Classification Schemas

This section characterises the main elements and entities one will find in a typical VRP and, consequently, in a VRP with synchronisation. It also serves as a glossary for the nomenclature that is later used throughout the paper. These concepts are instrumental to the classification schema presented in the second subsection, where each problem aspect is categorised.

2.3.1 Conceptual Schema

The VRP is a problem whose purpose is to outline the optimal sequence of tasks to be performed by a set of vehicles, given that each task must be performed at a given location. Building on this, the VRP is composed of at least two real-life entities: locations and vehicles.
Locations are *real-world sites that exist for performing tasks*. **Vehicles** are *real-world entities whose purpose is to perform routes*. In turn, **routes** consist of an *ordered sequence of tasks that need to be performed by a vehicle*.
Depending on the problem application, other entities may be considered equivalent to vehicles in a routing problem, such as machinery, people or staff. A vehicle usually performs

a single route, meaning that these two terms – vehicles and routes – can usually be used interchangeably.

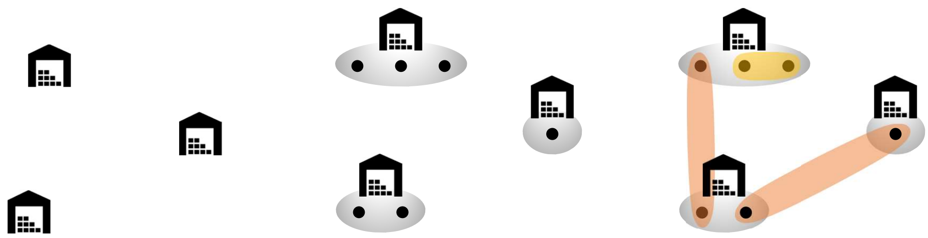
Resources are *entities of limited availability, available for consumption by one or more vehicles in order for them to fulfil tasks*. As defined in Drexl [2012], the main characteristic of a resource consists in its limited availability, which triggers a need for tasks, operations or vehicles in a problem to "compete" for its utilisation.

Depending on the specific problem requirement, the availability of a resource can be restricted to certain entities of the problem, such as a given location(s), route(s), or across the whole routing problem. Examples of resources include a route's available service time to fulfil tasks, the capacity of a vehicle (whose availability is also restricted to the specific route), and the demand of a location (whose availability is restricted to the specific location), among others.

In many routing problems, the terms task and location are used interchangeably, usually because the problem assumptions state that there is no more than one task in a given location (i.e., locations are only visited once). However, for the sake of generalisation, it is necessary to distinguish these two terms.

Tasks consist of *duties, usually mandatory, that must be done at a given location and require exactly one vehicle and zero or more units of some resource(s) to perform it*. In a standard VRP, each location has a single task, which consists in visiting it, while in a split delivery VRP, for example, locations would have more than one task associated with it.

For the purpose of this paper, we also present an additional definition that builds on the concept of a task. **Operations** consist of *associations of two tasks that must be performed by one or more vehicles at one or more locations and require zero or more units of some resource(s).* An operation is an abstraction of the task concept, used to link different tasks that are dependent in some form. The use cases for this concept are several, such as associating pickup and delivery tasks for given merchandise or linking different tasks that must be performed within a given time offset. By our definition, any task being performed by more than one vehicle is not a task but instead an operation composed of tasks. Also, any association of more than two tasks is not, by our definition, an operation; however, these associations can be split into two or more operations with two tasks each. Figure 2.4 visually illustrates the distinction between the concepts of location, task and operation, complementing the presented definitions.



(a) locations (b) tasks (c) operations

Figure 2.4: Illustration of the location, task and operation concepts

When a certain problem aspect or characteristic is introduced to a routing problem, it necessarily triggers interdependencies between entities of the problem, which need to be taken into account to obtain a feasible routing plan. Hence, we define **interdependencies** as the *relations established between different tasks, operations or routes of the problem where the feasibility of one depends on the feasibility of the other*. In other words, an interdependency exists between two VRP entities when it is not possible to guarantee that one remains feasible when the other is changed. Examples of interdependencies in the VRP include the ones triggered by typical vehicle capacity constraints – the fact that a given task is allocated to a vehicle may preclude other tasks from being allocated to that route due to capacity limits –, or synchronisation of tasks within given time offsets – the fact that a given task is performed at a given time limits the degrees of freedom for the time the other task will be performed.

It should be noted that the concept of interdependencies is not a synonym for synchronisation. However, the concept of interdependencies is pivotal to the classification schema that follows.

2.3.2 Classification Schema

The classification schema we present can take on every problem aspect of a routing problem and map it into one of four categories. The rationale behind this categorisation relies on two main assessments: (i) whether the problem aspect intends to link routing decisions with decisions from other optimisation problems; and (ii) given a solution to a routing problem where the problem aspect is currently infeasible, on which parts of the solution would a hypothetical solution algorithm need to intervene to get to the feasible domain.

In other words, one can say that this categorisation is linked with the direct scope of the interdependency that a problem aspect induces. The scope of an interdependency can be four-fold: within the same route, between routes, among the whole routing problem or between the routing problem and another optimisation problem. The first three interdependency scopes result in three routing problem aspect categories, while the latter does not involve interdependencies within the routing problem itself.

Figure 2.5 presents an overview of the classification schema and visually represents it.

The categories of the schema are:

- **Local Aspects** – concerns the problem aspects that, in the event a route is changed, only impact the feasibility of the route being changed; this category corresponds to what Irnich et al. [2014] would classify as “local” or “intra-route constraints”;

- **Synchronisation Aspects** – concerns the problem aspects that, besides impacting the route being changed, also impact other related routes; this category corresponds to the concept of interdependent routes presented by Drexl [2012];

- **Global Aspects** – concerns the problem aspects that, although they do not impact the individual feasibility of any route, affect the feasibility of the overall routing problem; this category corresponds to what Irnich et al. [2014] would classify as “global” or “inter-route constraints”;

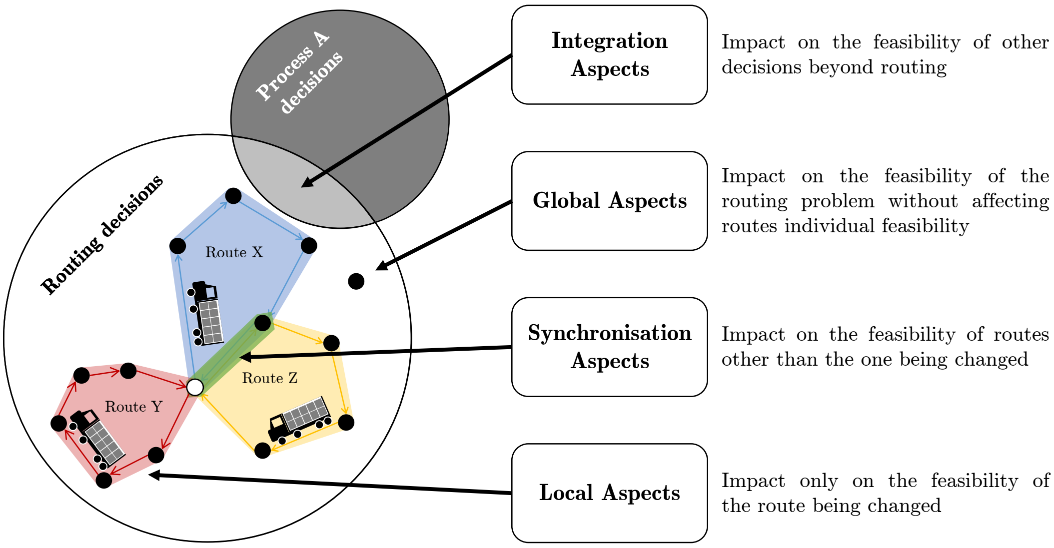


Figure 2.5: Visual representation of the proposed problem aspect classification framework

- **Integration Aspects** – concerns the problem aspects that can impact the feasibility of decisions beyond the scope of routing; in this category, we refer to problems where the interdependency is not established between the typical VRP entities, but between VRP entities and other external decisions, such as production or inventory decisions. The problem aspects that fit in this category are found in the general class of Integrated VRPs, considering the definition provided by Bektaş et al. [2015].

Each of the four categories contains several well-known problem aspects from the VRP literature. Figure 2.6 frames the main problem aspects into the previously presented categories.
The categories of this schema support the modelling framework presented in section 2.5, conceived in a modular fashion, thus allowing to obtain a general class of VRPs with synchronisation conjugated with other frequently found problem aspects.

2.3.3 Classifying a problem aspect

For analysing and determining how a given problem aspect should be classified, we provide a flowchart in Figure 2.7.
The flowchart starts by evaluating if the problem aspect under analysis links routing decisions with decisions from other optimisation problems, in which case, the aspect is classified as an integration aspect. If that is not the case, we progress through the flowchart and consider the following assumptions:

- **Assumption 1** – a feasible solution is known for a routing problem containing the problem aspect that is under analysis;
- **Assumption 2** – a given route is changed from the solution in such a way that all problem aspects remain feasible except for the one being analysed. Therefore, the

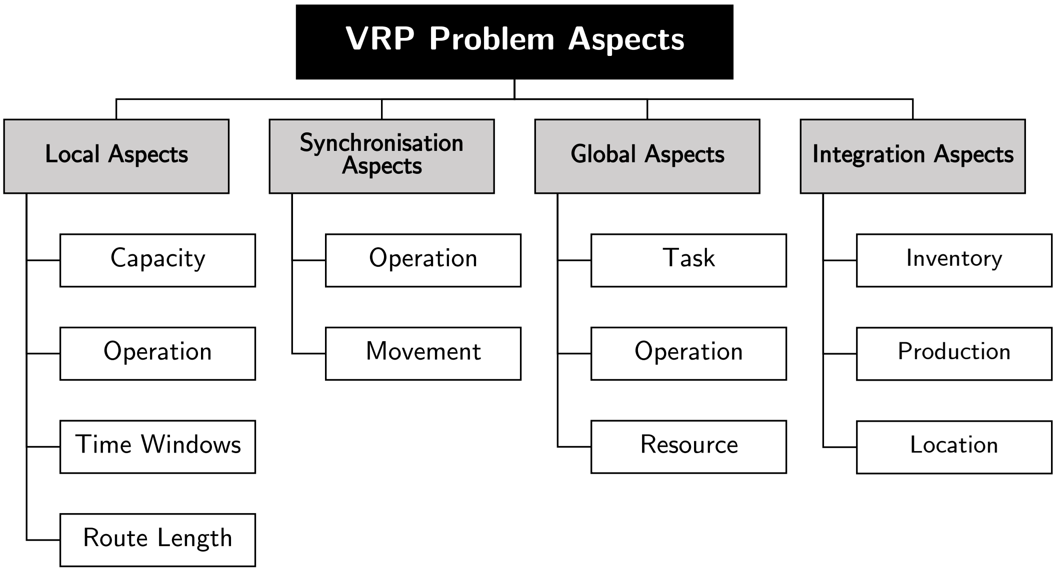


Figure 2.6: Main VRP problem aspects framed into the categories of the classification schema

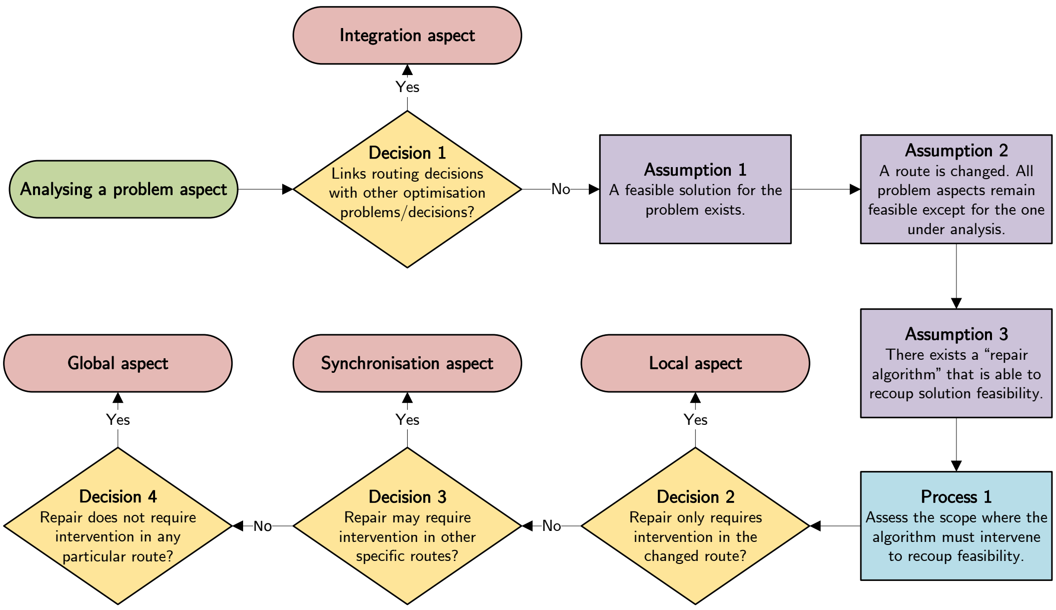


Figure 2.7: Flowchart for the classification of problem aspects

solution must be changed back to the feasible domain by ensuring the feasibility of that problem aspect;

- **Assumption 3** – there exists a “repair algorithm” that is able to “repair” the solution and make the problem aspect feasible again.

The described procedure starts by checking if the problem aspect under analysis only links decisions within the routing realm. If that is not the case, i.e., the problem aspect links routing decisions with decisions from other optimisation problems, we are in the presence of an integration aspect.

Progressing through the flowchart, if the problem aspect cannot be considered an integration aspect, it must be classified based on where the repair algorithm must intervene to restore its feasibility – henceforth called the “burden of repair”.

If the repair algorithm only needs to intervene in the route that was changed – i.e., the burden of repair resides in the route that was changed – it is considered a local aspect. This is the case for vehicle capacity constraints, for example, since, in this case, the hypothetical repair algorithm would only need to change the route where the capacity infeasibility occurred.

If the repair algorithm may also require intervention in other specific routes besides the one that was changed – i.e., the burden of repair extends to other specific routes besides the route being changed –, it is considered a synchronisation aspect. This is the case for temporal precedence constraints occurring between different vehicles, for example, since, in this case, the repair algorithm may need to not only intervene in the route that was changed but also change routes that are related to it.

If, instead, the repair algorithm needs to intervene in one or more routes but no one in particular – i.e., the burden of repair does not reside in any particular route –, the aspect is classified as a global aspect. This is the case for aspects ensuring that a given customer must be visited once by some vehicle, for example. Indeed, if this problem aspect is infeasible, the repair algorithm needs to intervene in some route, although there is no particular route that must be changed to return to feasibility. It is a different kind of interdependency than the one produced by synchronisation since the burden of performing the repair does not lie with any particular route.

2.3.4 Comparison with existing schemas in the literature

As it was previously stated, the focus of this work resides in the synchronisation aspects of VRPs. Nevertheless, the classification schema we propose applies to the general class of VRPs and not specifically to VRPs with synchronisation. The motivations for this approach are two-fold. First, it enables the contextualisation of synchronisation among other problem categories. Second, since our definition of synchronisation is more restrictive than other ones present in the literature, we are able to classify other problem aspects that we do not consider to be synchronisation into their corresponding categories, thus eliminating ambiguities that may arise.

The proposed schema builds on nomenclature and previously existing concepts of the literature. However, some fundamental differences exist between our proposed schema and

others in the literature, which are explained as follows.
Analysing the differences between our schema and the one in Irnich et al. [2014], we verify that Irnich et al. [2014] consider synchronisation a part of global problem aspects. In light of our criteria for our proposed schema, and because global aspects consist of problem requirements that are not indexed to a given route – unlike synchronisation aspects –, we consider that the scope of interdependencies between synchronisation and global aspects are so fundamentally different that synchronisation should be considered in its own category.
The proposed classification schema also acknowledges some fundamental differences in the types of problem aspects that are considered to be synchronisation. In fact, Drexl [2012] considers five synchronisation aspects – task, resource, load, operation and movement synchronisation. The proposed schema only considers two synchronisation aspects – operation and movement synchronisation – and does not consider the remaining three.
Actually, taking into account its underlying definitions and the criteria established for this schema, task, resource and load aspects should be classified as global aspects instead of synchronisation aspects. Furthermore, one can argue that these three problem aspects are actually present in most VRPs, even those without synchronisation. In fact, the task, resource and load synchronisation concepts presented by Drexl [2012] consider problem requirements that are not intrinsic to any particular routes and therefore do not establish a direct inter-route interdependency. Instead, the interdependency is established through the acknowledgement of trade-offs between different routing options or, in other words, inter-route competition for a given shared resource.
Overall, despite these fundamental differences of this proposed classification schema from others found in the literature, it is the opinion of the authors that this schema can further focus the concept of synchronisation in vehicle routing, thus providing a narrower and less ambiguous scope of the concept.
The following subsections will define each of these categories and problem aspects in further detail. To that effect, representative examples for each type of problem aspect are given. For each example of problem variant, an indication of the problem aspects is provided, as well as examples of references addressing these problems. These references may be the reference where the problem was introduced, a relevant review on the problem or another representative seminal reference. In Appendix 2.A of this paper, the problems addressed in all reviewed papers (including the given examples) are classified according to our proposed classification schema.

2.3.5 Local Aspects

These problem aspects comprise the ones that trigger an *interdependency between tasks of the same route*: in the event that a given route is changed, either by resequencing, removing or adding tasks to a route, the feasibility impact of this problem aspect is confined only to the route that was changed.
The main local aspects that are predominant in the literature are:

- *Capacity constraints*, which consist in interdependencies between tasks of the same route that are triggered by the capacity limits of vehicles;

- *Operation constraints*, which consist in interdependencies between tasks of the same route that are triggered by task sequencing or time offset requirements; examples of these constraints are found in the Pickup and Delivery Problem (PDP) [e.g., Parragh et al., 2008], as it is necessary that, within the same route, the pickup task of a specific request be performed before performing its corresponding delivery task;

- *Time Window constraints*, which consist in interdependencies between tasks of the same route that are triggered by the need to have a vehicle arrive at a task within a given time;

- *Route length constraints*, which consist in interdependencies between tasks of the same route that are triggered by the need for vehicle routes to have a limited duration/length.

Table 2.1 provides examples of different VRP variants one may find in the literature. For each problem variant, an indication of the problem aspects is provided, as well as examples of references addressing these problems.

Table 2.1: Examples of different types of local problem aspects

Problem variant / Application	**Representative Examples**	**Local Aspects**			
		Cap	***Op***	***TW***	***RL***
Capacitated VRP	Laporte et al. [1986]	•			
Pickup and Delivery Problem	Parragh et al. [2008]	•	•		
VRP with Backhauls	Deif and Bodin [1984], Goetschalckx and Jacobs-Blecha [1989]	◦	•		
VRP with synchronised pickup and delivery	Gschwind [2018]	◦	•	◦	
VRP with Time Windows	Savelsbergh [1992]		◦	•	
Distance-constrained VRP	Laporte et al. [1984]			◦	•

Legend: • - distinctive problem aspect for the problem variant/application; ◦ - problem aspect frequently found but not distinctive for the problem variant/application; Op - Operation constraints; Cap - Capacity constraints; TW - Time Window constraints; RL - Route Length constraints.

2.3.6 Synchronisation Aspects

Synchronisation aspects, the focal topic of this paper, concern the set of problem aspects that may impact the feasibility of routes other than the one that is being changed. They trigger an *interdependency between tasks of different routes*. In the event that a given route is changed by any means, these problem aspects may not only impact the feasibility of the route being changed but also of other routes that are "linked" with it. An example of synchronisation aspects consists in the problem requirements for the transportation of trailers by tractors in the Truck and Trailer Routing Problem (TTRP) [e.g., Chao, 2002], as a change in a trailer route can have effects on the route of the tractor that was originally transporting it.

The main synchronisation aspects can be summarised in two main categories, based on the nomenclature proposed by Drexl [2012]:

- *Operation synchronisation* – consists in the interdependencies between tasks of different routes that need to be performed within some temporal offset;
- *Movement synchronisation* – consists in the interdependencies between sequences of tasks among different routes.

Given the scope of this paper, we approach these synchronisation aspects in further detail, supported by Table 2.2, which provides an overview of examples and different variants of synchronisation aspects in the VRP literature.

Table 2.2: Examples of different types of synchronisation aspects

Problem variant / Application	**Representative Examples**	**Synchronisation Aspects**	
		Op	***Mov***
Home Health Routing and Scheduling Problem	Bredström and Rönnqvist [2008], Mankowska et al. [2013]	•	
Maritime shipping	Bakkehaug et al. [2016], Bélanger et al. [2006]	•	
Truck and Trailer Routing Problem	Chao [2002], Parragh and Cordeau [2017]	◦	•
Active-Passive VRP	Meisel and Kopfer [2012], Tilk et al. [2018]	◦	•

Legend: • - distinctive problem aspect for the problem variant/application; ◦ - problem aspect frequently found but not distinctive for the problem variant/application; Op - Operation constraints; Mov - Movement constraints.

2.3.6.1 Operation synchronisation

The purpose of operation synchronisation is to ensure that certain tasks are being performed within certain time limits. It can be formally defined as follows:

Consider an operation (a,b). If its composing tasks a and b are being performed by two different routes k and k', respectively, then they must be performed within a given temporal offset.

Operation synchronisation is useful for certain complex operations that require more than one vehicle to visit a certain customer to accomplish it successfully.
A problem that acknowledges operations synchronisation is the Home Health Care Routing and Scheduling Problem (HHCRSP) [e.g., Bredström and Rönnqvist, 2008, Mankowska et al., 2013]. Other typical applications concern aircraft scheduling or vessel scheduling at ports [e.g., Bélanger et al., 2006, Bakkehaug et al., 2016]. The most representative examples of operation synchronisation are described in further detail in section 2.4.

In our classification schema, operation synchronisation concerns only the need to temporally synchronise a given operation when its composing tasks are performed by different vehicles. Contrary to the definition of operation synchronisation of Drexl [2012], other additional requirements, such as ensuring that each task of the given operation is performed, are not significantly different from other problem aspects not directly related to synchronisation. Therefore, they are not considered here, and they are mapped to other problem aspects in our classification schema.

2.3.6.2 Movement synchronisation

The formal definition of movement synchronisation is the following:

Consider an operation (i, j), composed of tasks i and j, henceforth called passive operation. Let us also consider a set of n operations $\{(a_1, b_1), ..., (a_n, b_n)\}$, henceforth called active operations. If a route performs task j immediately after task i, then there must exist at least one route that performs task b immediately after task a, with (a, b) belonging to the set of active operations.

Movement synchronisation concerns the inter-route dependency between two vehicles whose task sequences need to be synchronised. Movement synchronisation is a frequent requirement with routing problems acknowledging vehicles of different types, namely vehicles that cannot move autonomously and vehicles that need to transport these non-autonomous vehicles [Drexl, 2012].

Typically, movement synchronisation occurs between passive and active operations. A passive operation is one that requires that an active operation be performed in order for it to be performed as well. In turn, an active operation is one that may be accompanied or not by a passive operation.

The concepts of active and passive operations can be easily translated into different problem variants from the literature, although usually, these problem variants apply these concepts to the vehicles themselves rather than the tasks being performed, as explained below. For the sake of this schema, vehicles are not classified as active or passive. In fact, a vehicle could exhibit both active and/or passive behaviour in different points of its route. Instead, the active/passive attribute should be associated with operations, depending on the movement synchronisation constraints being enforced.

The most typical problem variants with movement synchronisation are the Truck and Trailer Routing Problem (TTRP) [e.g., Chao, 2002, Parragh and Cordeau, 2017] and the Active-Passive VRP [e.g., Meisel and Kopfer, 2012, Tilk et al., 2018].

In our classification schema, movement synchronisation concerns only the need to ensure that, when traversed by different vehicles, a given pair of arcs is traversed. The classification schema of Drexl [2012] defined movement synchronisation as the simultaneous traversal of arcs, which maps into our proposed schema through both movement and operation synchronisation. In most real-life applications, movement synchronisation typically implies that operation synchronisation is also performed, as it is necessary that the arcs be traversed simultaneously. However, in a general context, this may not always be the case.

2.3.7 Global Aspects

Global aspects concern the ones that do not directly impact the individual feasibility of each one of the routes of the problem, but in turn may impact the overall feasibility of the routing problem. This occurs because the *interdependencies occur between routes but are triggered by the availability of a resource* that is not associated with a specific vehicle.
An example of a global aspect is the "all nodes are visited exactly once" requirement, which is intrinsic to standard VRPs. In fact, if a given node is removed from a route, this problem requirement still allows all routes to remain feasible individually, although this problem requirement is not satisfied. This happens because global aspects are not intrinsic to any specific route but are a requirement that needs to be considered in an aggregated manner and not in a per-route sense.
The global aspects presented in this paper are the following:

- *Task constraints* – consist in the interdependencies between routes triggered by the need to have a vehicle satisfying a given task – they establish the requirement for having a given task performed at least once, exactly once and/or by a suitable vehicle;
- *Operation constraints* – consist in the interdependencies between routes triggered by the need to have a given operation performed – they establish whether an operation must/can be performed by the same vehicle or by different vehicles;
- *Resource constraints* – consist in the interdependencies between routes triggered by the scarcity of a resource that is accessible/consumable by multiple vehicles.

Table 2.3 provides an overview of examples and different variants of global aspects in the VRP literature.

2.3.8 Integration Aspects

Integration aspects concern the problem aspects that trigger *interdependencies between the routing decisions and decisions concerned with other planning processes*. Therefore, this set of problem aspects refers to all the trade-offs and feasibility issues that do not directly concern routes, but rather the integration of routing decisions with other planning problems. These problem aspects translate into problem variants that are called in the literature Integrated VRPs (iVRPs), which consist of VRPs that are linked with other optimisation problems. Several practical examples of iVRPs can be found in the literature, such as inventory or production routing [e.g., Neves-Moreira et al., 2019, Cordeau et al., 2015, Adulyasak et al., 2015].
Distinguishing between intra-process and inter-process interdependencies is a distinguishing feature of this classification schema, which, to the best of our knowledge, has not been properly contextualised in the literature.

Table 2.3: Examples of different types of global problem aspects

Problem variant / Application	Representative Examples	Global Aspects		
		Tsk	*Op*	*Res*
Home Health Care Routing and Scheduling Problem	Bredström and Rönnqvist [2008], Mankowska et al. [2013]	•	•	
Maritime shipping	Bakkehaug et al. [2016], Bélanger et al. [2006]	•	•	
Truck and Trailer Routing Problem	Chao [2002], Parragh and Cordeau [2017]	○	•	
Active-Passive VRP	Meisel and Kopfer [2012], Tilk et al. [2018]	○	•	
VRP with Profits	Chu [2005], Archetti et al. [2009]	•		
Split Delivery VRP	Archetti and Speranza [2012], Irnich et al. [2014]	•		•
Mixed Fleet VRP	Baldacci et al. [2008]	•		•
VRP with Cross-Docking	Petersen and Ropke [2011], Yu et al. [2016]	○		•

Legend: • - distinctive problem aspect for the problem variant/application; ○ - problem aspect frequently found but not distinctive for the problem variant/application; Tsk - Task constraints; Op - Operation constraints; Res - Resource constraints.

2.4. Overview of the State of the Art

In this section, an overview of the state of the art on the VRP with synchronisation is presented by resorting to the outputs provided by the integrative literature review, whose complete results are presented in the form of a table in Appendix 2.B. For each reference of the literature review, the table indicates the predominant usage of synchronisation, the problem application, the objective function components being optimised, the categories of problem aspects found, the solution method approaches, and well as an indication of the sources of uncertainty and dynamism (if any). In the first subsection, a general description of the predominant routing problems with synchronisation is performed, followed by an overview of the adopted solution approaches. The section concludes with an analysis of routing problems with synchronisation subject to uncertainty and dynamism.

2.4.1 Predominant routing problems with synchronisation

Routing with synchronisation of schedules. A significant part of the publications found tackle problems where the focus of the problem is not so much on determining a routing plan for a given fleet of vehicles, but rather on obtaining a feasible schedule for a given set of tasks that need to be performed, subject to operation synchronisation.

The most representative problem that matches these characteristics is the Home Health Care Routing and Scheduling Problem (HHCRSP) [e.g., Bredström and Rönnqvist, 2008]. These problems are typically characterised by a fleet of staff elements that must serve a set of patients (customers) by performing certain tasks/services at their homes. Some of these services require more than one staff person to be present at the customer location, thus requiring synchronisation. The operation synchronisation requirements can be either of simultaneity (e.g., staff members must perform their tasks simultaneously) or precedence (e.g., a task must be performed by a given staff member before a second task is started). In these problems, other aspects are typically found alongside operation synchronisation. On local aspects, time window constraints are frequently found [e.g., Decerle et al., 2018], while on global aspects, task constraints are usually required to ensure that each task is performed by an appropriate staff member [e.g., Mankowska et al., 2013, Masmoudi et al., 2018], and operation constraints are required to ensure that interrelated tasks are performed by different staff members.

Similar applications of scheduling-focused routing problems with synchronisation can be found in the literature. The public utilities sector is another instance where synchronisation of schedules is applied. Goel and Meisel [2013] address a routing problem in the electricity sector, whose purpose is to schedule and synchronise maintenance operations at multiple points of an electricity network. Hà et al. [2020] address a similar scheduling problem in the scope of an internet service provider. Ali et al. [2021], Hanafi et al. [2019], Hojabri et al. [2018] tackle a routing problem with synchronisation for the delivery and installation of large items to final customers, where the installation vehicles must arrive at the customer after the items have been delivered by another vehicle. Scheduling with synchronisation issues are also prevalent in problems involving the transportation of perishable items [e.g., Anaya-Arenas et al., 2019] or railway maintenance [e.g., Pour et al., 2019].

The synchronisation of schedules is also especially relevant in routing problems that rely on the efficient use of scarce resources. In these applications, it is common to use synchronised scheduling between vehicles to ensure that these resources are not being utilised simultaneously. Bakkehaug et al. [2016] present a problem in the field of ship routing, whose purpose consists in maintaining capacity at ports feasible throughout time, therefore obtaining vessel schedules that ensure vessel departures and arrivals occur within predetermined time offsets. Fedtke and Boysen [2017] present a synchronised scheduling problem in a modern rail-rail transhipment yard, requiring the synchronisation of gantry cranes and shuttle cars. Froger et al. [2022] apply schedule synchronisation to a routing problem involving electric vehicles, where the arrival times to charging stations must be synchronised. There are various other applications considering the synchronisation of resource schedules, such as in the construction sector [e.g., An et al., 2018, Grimault et al., 2017], log-truck scheduling [e.g., Hachemi et al., 2013] or in humanitarian and military logistics [e.g., Lam and Hentenryck, 2016].

The objective functions of routing problems involving the synchronisation of schedules can vary greatly. Within the home health care application, one of the most common objective function components for this problem variant is the maximisation of vehicle-customer preferences [e.g. Afifi et al., 2015, Ait Haddadene et al., 2019, Haddadene et al., 2016], as an attempt to improve service quality to patients providing them with the caregivers they most prefer. In a more general manner, routing problems with synchronisation of schedules also consider the minimisation of scheduling-related components, such as the total travel time [e.g., López-Aguilar et al., 2018, Polnik et al., 2020, Fikar et al., 2016a, Frifita and

Masmoudi, 2020], the total waiting time [e.g., Doulabi et al., 2020] or even the total tardiness [e.g., Mankowska et al., 2013]. Other less common objectives can be found, such as the maximisation of the number of performed tasks [e.g., Dohn et al., 2009]. Nevertheless, the classical objective of minimising the total distance costs is still frequently found [e.g., Cappanera et al., 2020, Ghilas et al., 2016b, Parragh et al., 2008, Rasmussen et al., 2012], as well as the minimisation of vehicle fixed costs [e.g., Liu et al., 2019, Nguyen et al., 2013].

Routing with transfers or cross-docking requirements. The literature review also shows that synchronisation is common in routing problems with transfers or cross-docking requirements. In these types of problems, transfer locations exist, where it is necessary to ensure operation synchronisation between the drop-off and collection of the request at the transfer/cross-docking location by different vehicles. Furthermore, it is typically necessary to consider the capacity limits of each vehicle.

The PDP with Transfers (PDPT) is a typical problem variant that fits into this description [e.g., Masson et al., 2013b, Peng et al., 2019, Qu and Bard, 2012, Rais et al., 2014], where vehicles must perform pickup and delivery requests, but the pickup task may not be necessarily performed by the same vehicle that will deliver it. In this context, operation synchronisation is necessary at transfer locations so that the drop-off of a request occurs before its collection by another vehicle.

The VRP with Cross-Docking [e.g., Grangier et al., 2017, 2019, Yin and Chuang, 2016] is another prevalent problem variant in this context. In this problem, cross-docking locations can be used to transfer load between vehicles. Usually, operation synchronisation is necessary to establish temporal precedence between the unloading and loading tasks of different vehicles. Similarly to what happens with the VRP with Cross-Docking, multiple-echelon VRPs are also a predominant variant [e.g., Dellaert et al., 2021, Grangier et al., 2016, Li et al., 2021, Marques et al., 2020, Medina et al., 2019, Mirhedayatian et al., 2019, Nolz et al., 2020], which also involve operation synchronisation in order to take into account the temporal precedence relationships between routes of consecutive echelons of the routing network.

Although less common, other problem variants with transfers or cross-docking requirements can be found in the literature, such as the Dial-a-Ride Problem (DARP) with Transfers [e.g., Masson et al., 2014], for people transportation, or the Line-haul Feeder VRP [e.g., Brandstätter and Reimann, 2018, Brandstätter, 2019].

Besides synchronisation, these problems are typically subject to several local and global aspects. On local aspects, capacity constraints are usually a requirement [e.g., Rais et al., 2014]. On global aspects, task constraints are essential for ensuring that all mandatory tasks are performed, as well as operation constraints, so as to determine whether a pickup/delivery operation can/must be performed by the same vehicle or by different vehicles.

In general, the objective functions of routing problems involving transfers or cross-docking operations are not distinctively different from traditional objective components, namely the minimisation of distance costs [e.g., Grangier et al., 2017, Nolz et al., 2020] or the minimisation of vehicle fixed costs [e.g., Grangier et al., 2016, Qu and Bard, 2012]. Still, some alternative objectives can be found, such as the minimisation of the total route duration [e.g., Brandstätter and Reimann, 2018, Brandstätter, 2019], the minimisation of waiting times [e.g., Li et al., 2021] or even the minimisation of the makespan [e.g., Salazar-Aguilar et al., 2013].

Routing with autonomous vehicles. The most representative problem that acknowledges synchronisation with autonomous vehicles is the VRP with Drones (VRP-D) [e.g., Chung et al., 2020]. It is a routing problem with two different types of vehicles – trucks and drones – where the drone is an auxiliary vehicle to the truck, which can perform small-size deliveries. Due to its limited autonomy, the drone can only travel up to certain distances, and it must return to the truck for charging or picking up new requests to be delivered.
Although they are autonomous, the limited autonomy of drones requires that they may need to be "parked" on the truck for charging while the truck is performing its route. In these situations, the truck and drone are subject to movement synchronisation, as they are moving simultaneously.
Local aspects and global aspects can also be found in these types of problems. On local aspects, operation constraints are required to ensure that autonomous vehicles respect the required tasks precedences in performing a given task. Furthermore, route length constraints are also essential to establish that the limited autonomy of the drone is not exceeded. On global aspects, task constraints are necessary for having each task being performed by an appropriate vehicle (a truck, an autonomous vehicle, or either one of them).
The emergence of routing problems with synchronisation considering autonomous vehicles is fairly recent and has become a hot topic, especially for the e-commerce sector as a way to optimise last-mile delivery to final customers [e.g., Agatz et al., 2018, Dayarian et al., 2020, Mourad et al., 2021, Schermer et al., 2020].
The objective functions considered in these types of problems can be rather diverse. Besides the typical minimisation of distance costs [e.g., Agatz et al., 2018, Coindreau et al., 2021], common objective function components include the minimisation of travel times [e.g., Kitjacharoenchai and Lee, 2019, Kitjacharoenchai et al., 2020, Simoni et al., 2020], the minimisation of the total route duration [Dell'Amico et al., 2019, Kitjacharoenchai et al., 2019, Reed et al., 2022, e.g.,] or the minimisation of the makespan [e.g., Es Yurek and Ozmutlu, 2018, Murray and Chu, 2015]. Other less common objectives can be found, such as the minimisation of delays [e.g., Boysen et al., 2018] or the minimisation of waiting times [e.g,. Li et al., 2020a].

Routing with trailers or passive vehicles. The archetypal problem acknowledging trailers or passive vehicles is the Truck and Trailer Routing Problem (TTRP) [Chao, 2002]. The problem consists in determining the routes for a set of trucks that may have a trailer coupled to them. Trailers allow a reduction of the number of routes needed to perform deliveries to customers by increasing transportation capacity; however, due to site-dependent constraints, some customers are unable to be visited by trailers, so trucks must first visit a transhipment node, transfer load between the truck and trailer and uncouple. Afterwards, the truck can perform sub-tours to trailer-incompatible customers. Another problem variant one may find in the literature is the Active-Passive VRP (APVRP) [Meisel and Kopfer, 2012]. One of the applications of this problem is in drayage operations, where active vehicles (trucks) are responsible for transporting passive vehicles (containerised cargo) from a pickup to a delivery location, taking into account that there is no pre-defined assignment between trucks and trailers, unlike the TTRP.

Movement synchronisation is the most relevant problem aspect of these problems since it is necessary to ensure that a passive vehicle can only transit to another task if there is also an active vehicle transiting. Along with movement synchronisation, operation synchronisation is also required to ensure that movement synchronisation between the active and passive vehicles occurs simultaneously.

On what local aspects are concerned, capacity constraints are essential for ensuring that vehicle capacity is not exceeded, and operation constraints are also necessary for limiting the tasks that can be performed after another one is complete. On global aspects, task constraints ensure that each task is performed by compatible vehicles, and operation constraints are also necessary for ensuring that given operations must necessarily be performed by different vehicles.

Other applications were found for routing problems acknowledging passive vehicles. Examples include the Driver and Vehicle Routing Problem [e.g., Domínguez-Martín et al., 2018b], whose goal is to not only schedule tasks but also assign drivers to the vehicles throughout the day, therefore arising the need for drivers to be subject to movement synchronisation with vehicles.

The minimisation of distance is the most predominant objective function component being optimised in these routing problems [e.g., Xue et al., 2014, Yakıcı et al., 2018], followed by the minimisation of vehicle fixed costs [e.g., Drexl, 2014]. Other less common objectives include the minimisation of the makespan [e.g., Hu and Wei, 2018, Salazar-Aguilar et al., 2012], the minimisation of delays [e.g., Fink et al., 2019, Furian et al., 2018] or the minimisation of route duration [e.g., Meisel and Kopfer, 2012, Soares et al., 2019].

2.4.2 Solution approaches

The literature on solution methods for the VRP with synchronisation is quite diverse and is still in an exploratory phase, probably because no consensual method has been found that systematically outperforms others.

One of the first works that resorted to exact methods for a VRP with operation synchronisation was Dohn et al. [2011], which proposed Dantzig-Wolfe decompositions to allow for column generation approaches. Since then, other column generation approaches were envisaged [e.g., Dellaert et al., 2021, Emadikhiav et al., 2020, Li et al., 2020b, Luo et al., 2016, Rothenbächer et al., 2018, Tilk et al., 2019], most of them only referring to problems with operation synchronisation. References addressing problems that contain movement synchronisation aspects and resort to column generation are less frequent [Fink et al., 2019, Parragh and Cordeau, 2017, Rothenbächer et al., 2018, Tilk et al., 2018]. In parallel with column generation, a significant part of the above-cited references also resorts to the generation of cutting planes. Standalone branch-and-cut approaches are also commonly found in the literature, both for operation synchronisation [e.g., Bianchessi et al., 2019, Hanafi et al., 2019, Doulabi et al., 2020, Luo et al., 2016] and movement synchronisation [e.g., Domínguez-Martín et al., 2018a,b, Drexl, 2014, Schermer et al., 2020, Tamke and

Buscher, 2021].
However, the most predominant solution approaches for routing problems with synchronisation consist in heuristic concepts. Large Neighbourhood Search (LNS) is the most popular of them all, probably due to the general popularity of this approach in routing problems, especially of its adaptive version presented by Pisinger and Ropke [2007]. LNS approaches usually use well-known removal and insertion operators from the literature [e.g., Parragh and Doerner, 2018, Liu et al., 2019, Roozbeh et al., 2020], such as Worst Removal, Shaw Removal, Best Insertion or k-Regret Insertion. However, the interdependence problem of synchronisation typically constitutes a non-negligible factor that may generate infeasible solutions, which in an LNS framework requires workarounds for either "repairing" solutions on each iteration or evaluating the feasibility of the candidate. For example, Ghilas et al. [2016a] propose an Adaptive LNS for a rich pickup and delivery problem with transfers, where after a removal and insertion iteration, a repair procedure is invoked to account for transfer requirements. A similar procedure is adopted for Bakkehaug et al. [2016] in a routing problem with voyage separation requirements. Coindreau et al. [2021] used a different approach for addressing the interdependence problem, which consisted in an efficient feasibility check procedure inspired by Masson et al. [2013a]. The LNS approach is also frequently found for VRPs with movement synchronisation, especially in applications involving unmanned aerial vehicles [e.g., Kitjacharoenchai et al., 2020, Li et al., 2020a, Schermer et al., 2020].
Some works involving movement synchronisation also use other metaheuristic concepts such as Variable Neighbourhood Search [e.g., Coindreau et al., 2019, Ritzinger et al., 2017], Simulated Annealing [e.g., Chao, 2002] or Tabu Search [e.g., Chao, 2002]. In works involving operation synchronisation, population-based heuristics can also be found [e.g., Ait Haddadene et al., 2019, Huang et al., 2018], although they are less common. Matheuristic approaches are also rarely found [e.g., Bredström and Rönnqvist, 2008, Soares et al., 2019], which indicates that further work may be envisaged on this front.
The solution approaches that are used for each reference in the literature review can be found in the extensive table in Appendix 2.B.

2.4.3 Synchronisation under uncertainty and dynamism

Another future challenge for the VRP with synchronisation consists in the real-life implementation of routing plans with vehicle synchronisation and its reactivity to unforeseen events or routing requests.
While for traditional VRPs, dynamic routing usually needs only to reconstruct portions of the initial solution when faced with an unexpected event or request because routes are independent, this is not the case for the VRP with synchronisation. It is expected that, as more synchronisation aspects are introduced to a problem, the more changes a reconstructed routing solution will have compared to the initial plan. The literature on dynamic approaches for VRPs with synchronisation is scarce. Rousseau et al. [2013] study a dynamic VRP with operation synchronisation. A constraint programming model is envisaged, which is then used for implementing an insertion heuristic to include new customers into the initial solution. The work acknowledges the increasing difficulty when trying to insert

new customers that require synchronisation. Fikar et al. [2016b] present a discrete-event driven metaheuristic for a VRP with both operation and movement synchronisation. Its application is based on home health care routing with combined trip sharing and walking. Home carers may move between locations by foot (if the destination is within walking distance) or be transported by a vehicle that can transport multiple carers. The solution algorithm iteratively generates new solutions based on the occurrence of events over time, such as the pickup or delivery of carers by the vehicles or the assignment of the next customer to a home carer. More recently, Dayarian et al. [2020] present a VRP with drone resupply, where orders arrive dyna mically: the number of locations to serve may not be known when the vehicle departs from the depot, and therefore the routes need to be periodically re-optimised.

Similarly to what happens with dynamic routing, routing with sources of uncertainty is also unexplored. Literature of VRPs with synchronisation considering problem parameters as random variables, such as service times or travel times between locations, is very limited. Furian et al. [2018] consider the home health care routing case with operation synchronisation and propose an optimisation procedure embedded in a discrete-event simulation framework that handles stochasticity. Also, in the field of health care, Doulabi et al. [2020] consider a VRP with synchronised visits and stochastic travel and service times. Medbøen et al. [2020] consider uncertain travel times triggered by harsh weather conditions in a maritime transportation problem, also resorting to a simulation-optimisation framework. Shi et al. [2020] present a routing problem with synchronised operation synchronisation considering greenhouse gas emissions, which resorts to robust optimisation for two uncertainty sets of travel and service times. In the context of city logistics, Anderluh et al. [2019a] use a Monte Carlo simulation with an optimisation approach for a 2-echelon VRP involving the synchronisation of vans and bicycles at satellite locations considering uncertain travel times.

Stochastic VRPs with synchronisation have already been acknowledged as a research gap that should be addressed as the topic becomes more mature [e.g., Parragh and Doerner, 2018].

2.5. Modelling framework

This section now provides the unified mathematical formulation for the VRP with synchronisation constraints, according to the previously described conceptual and classification schema. In order to model this problem without loss of generality, we will present a baseline mathematical formulation.

Modelling routing problems with different types of problem aspects represents a challenge, as it requires a flexible mathematical formulation to model different problem requirements. Therefore, the formulation that we will present will not focus on efficiency or scalability but rather on systematisation and consistency.

2.5.1 Sets and parameters

Let us consider a generic vehicle routing problem base formulation, whose purpose is to perform a set of n tasks, geographically dispersed through m locations, by means of a set of routes being performed by a set of K vehicles. The routes start and end in a depot through tasks 0 and $n+1$, respectively.
Especially when modelling synchronisation aspects in the same location, the problem's transportation network does not necessarily correspond to a real-life transportation network. For this mathematical formulation, we will continue with the nomenclature adopted in the previous section, and therefore we will use the term "tasks" to refer to vertices of the transportation network and use the term "locations" to refer to real-world locations.
It is assumed that each vehicle performs only one route, and therefore these concepts may be used interchangeably. However, routing problems where vehicles are able to perform multiple trips during the planning horizon can also be modelled with this framework by assuming that a vehicle route can have multiple legs that start and end in the depot. In this case, the depot location will have optional tasks for each vehicle, which are specifically designed for finishing and starting trips. For each of these tasks, the vehicle will be able to enter it to finish a trip, after which it will exit that task in order to start a new trip. The number of tasks that need to be generated for each vehicle under these conditions depends on the problem size, being $n-1$ a possible upper bound.
The problem sets are the following:

$\mathcal{L}$	Set of locations
$\mathcal{K}$	Set of vehicles or routes, $\mathcal{K} = \{k_1, k_2, ..., k_K\}$
$\mathcal{O}^k$	Set of depot tasks for finishing and starting possible trips of route $k \in \mathcal{K}$
$\mathcal{O}$	Set of all depot tasks for finishing and starting possible trips
$\mathcal{N}$	Set of tasks, $\mathcal{N} = \{1, 2, ..., n\} \cup \mathcal{O}$
$\mathcal{N}_0$	Set of tasks with depot tasks, $\mathcal{N}_0 = \mathcal{N} \cup \{0, \lvert\mathcal{N}\rvert + 1\}$
$\mathcal{A}$	Set of arcs $(i, j), \mathcal{A} \subset \mathcal{N}_0 \times \mathcal{N}_0$
$\mathcal{E}_i$	Set of tasks that can be performed immediately after task $i \in \mathcal{N}_0$
$\mathcal{R}$	Set of operations $(i, j), \mathcal{R} \subset \mathcal{N}_0 \times \mathcal{N}_0$

The total number of tasks in the problem, $\lvert\mathcal{N}\rvert$, can also be represented as N. The transportation network $\mathcal{G} = (\mathcal{N}_0, \mathcal{A})$ consists of a directed and incomplete graph of tasks that need to be performed. The set of arcs can be mathematically defined as presented in Equation (2.1).

$$\mathcal{A} = \{(i, j) \in \mathcal{N}_0 \times \mathcal{N}_0 : i \neq j, i \neq N+1, j \neq 0\} \tag{2.1}$$

Depending on the specific problem aspects or instance data (e.g., time window constraints, operation constraints, etc.), $\mathcal{A}$ may be subject to further preprocessing.
Let $\psi(i)$ be an auxiliary function that returns the location that is associated with task i.

The parameters of the problem are:

a_i, b_i	Earliest, latest possible time to begin performing task $i \in \mathcal{N}_0$
c_{ij}^k	Cost for traversing arc $(i, j) \in \mathcal{A}$ by vehicle $k \in \mathcal{K}$
d_{ij}	Travel distance from location $\psi(i)$ to location $\psi(j)$
t_{ij}	Travel time from location $\psi(i)$ to location $\psi(j)$
q_i	Demand to be satisfied for task $i \in \mathcal{N}_0$
r_i^k	takes value 1, iff task $i \in \mathcal{N}_0$ can be performed by vehicle $k \in \mathcal{K}$
s_i	Service time of task $i \in \mathcal{N}$
p_{ij}	takes value 1, iff $(i, j) \in \mathcal{R}$ is a passive operation
$z_{ij}^{i'j'}$	takes value 1, iff passive operation (i, j) can be performed with active operation (i', j')
Q^k	Capacity of vehicle $k \in \mathcal{K}$
δ_{ij}	Min. time offset between arrival times of task pairs/operation (i, j)
Δ_{ij}	Max. time offset between arrival times of task pairs/operation (i, j)
λ_{ij}	Min. time offset between the arrival times of synchronised operation (i, j)
μ_{ij}	Max. time offset between the arrival times of synchronised operation (i, j)
D^k	Maximum route length for route $k \in \mathcal{K}$
T	Planning horizon

It is assumed that travel costs respect the triangle inequality, meaning that $c_{ij}^k + c_{jl}^k \geq c_{il}^k, \forall i, j, l \in \mathcal{N}_0, k \in \mathcal{K}$. This assumption is equally applicable to travel distances d_{ij} and travel times t_{ij}. If multiple trips are acknowledged, it can be inferred that, for each vehicle k and each task i contained in O^k, parameter r_i^k will take value 1, and value 0 in all other instances.

2.5.2 Decision variables

We consider the following decision variables:

$$x_{ij}^k \quad \begin{cases} 1 & \text{if arc } (i, j) \in \mathcal{A} \text{ is traversed by vehicle } k \in \mathcal{K} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij}^{kk'} \quad \begin{cases} 1 & \text{if operation } (i, j) \in \mathcal{R}\text{, with tasks } i \text{ and } j\text{, is performed by routes } k \text{ and } k'\text{,} \\ & \text{respectively} \\ 0 & \text{otherwise} \end{cases}$$

u_{ij}^k	Load of vehicle $k \in \mathcal{K}$ when traversing arc $(i, j) \in \mathcal{A}$
w_i^k	Arrival time of vehicle $k \in \mathcal{K}$ at task $i \in \mathcal{N}_0$

Variables u_{ij}^k will only be necessary for capacitated VRPs, i.e. with interdependencies triggered by vehicle capacity and customer/task demand. On the other hand, variables w_i^k must always be present in the formulation, because its purpose is not only to be able to

model temporal interdependencies between tasks but also to guarantee the elimination of sub-tours if no other constraints ensure it.
The domain and nature of the decision variables used in this model are presented at the end of this modelling framework in constraints (2.37)–(2.40).

2.5.3 Objective function

The objective function of the base VRP problem will be the minimisation of the total travel costs (Eq. 2.2).

$$\min \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} c_{ij}^k x_{ij}^k \tag{2.2}$$

2.5.4 Fundamental routing constraints

The main constraints associated with this base model are the ones that follow.

$$\sum_{k\in\mathcal{K}} \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k = 1 \qquad \forall j \in \mathcal{N} \setminus \mathcal{O} \tag{2.3}$$

$$\sum_{k\in\mathcal{K}} \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k \leq 1 \qquad \forall j \in \mathcal{O} \tag{2.4}$$

$$\sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k - \sum_{i:(j,i)\in\mathcal{A}} x_{ji}^k = 0 \qquad \forall j \in \mathcal{N}, k \in \mathcal{K} \tag{2.5}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j}^k = \sum_{i:(i,N+1)\in\mathcal{A}} x_{i,N+1}^k \qquad \forall k \in \mathcal{K} \tag{2.6}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j}^k \leq 1 \qquad \forall k \in \mathcal{K} \tag{2.7}$$

$$w_j^k \leq T \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k \qquad \forall j \in \mathcal{N}_0, k \in \mathcal{K} \tag{2.8}$$

$$w_0^k + t_{0j} \leq w_j^k + T\left(1 - x_{0j}^k\right) \qquad \forall j : (0,j) \in \mathcal{A}, k \in \mathcal{K} \tag{2.9}$$

$$w_i^k + s_i + t_{ij} \leq w_j^k + T\left(1 - x_{ij}^k\right) \qquad \forall (i,j) \in \mathcal{A} : i \neq 0, k \in \mathcal{K} \tag{2.10}$$

$$\sum_{l:(l,i)\in\mathcal{A}} x_{li}^k + \sum_{l:(l,j)\in\mathcal{A}} x_{lj}^{k'} - 1 \leq y_{ij}^{kk'} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{2.11}$$

$$y_{ij}^{kk'} \leq \sum_{l:(l,i)\in\mathcal{A}} x_{li}^k \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{2.12}$$

$$y_{ij}^{kk'} \leq \sum_{l:(l,j)\in\mathcal{A}} x_{lj}^{k'} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{2.13}$$

$$x_{ij}^k \leq r_i^k r_j^k \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K} \tag{2.14}$$

Constraints (2.3) and (2.4) ensure that tasks must, or can, be performed exactly once, respectively. In this formulation, we assume there are no optional tasks except when multiple trips are allowed, in which case, depot tasks for finishing and starting new trips are optional. Constraints (2.5) establish the inflow and outflow conservation: vehicles entering a task node must also exit it. Constraints (2.6) and (2.7) ensure that every vehicle starts and ends its route at the depot. Constraints (2.8) are linking constraints between variables x_{ij}^k and w_i^k; they impose that the arrival time of vehicle k at task i cannot be different from zero if the vehicle does not perform said task $\left(x_{ij}^k = 0 \implies w_i^k = 0\right)$. Constraints (2.9) and (2.10) establish vehicle arrival times to task nodes. They also serve as sub-tour elimination constraints and are an alternative to the Miller-Tucker-Zemlin constraints. Constraints (2.11)–(2.13) are linking constraints that set the values of decision variables $y_{ij}^{kk'}$ based on the values of variables x_{ij}^k. Constraints (2.11) set variable $y_{ij}^{kk'}$ to 1 if tasks i and j of operation $(i, j) \in \mathcal{R}$ are being performed. Constraints (2.12) and (2.13) set the opposite cases. When task i is not being performed, constraints (2.12) forcefully set the value of $y_{ij}^{kk'}$ to zero. When task j is not being performed, constraints (2.13) forcefully set the value of $y_{ij}^{kk'}$ to zero. Constraints (2.14) are vehicle-task compatibility constraints: they establish that an arc (i, j) can only be traversed by vehicle k if tasks i and j can both be performed by k.

2.5.5 Local aspects

Capacity constraints. The inclusion of capacity constraints can be achieved by adding the constraints that follow and considering decision variables u_{ij}^k and parameters q_i and Q^k.

$$u_{ij}^k \leq Q^k x_{ij}^k \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K} \tag{2.15}$$

$$\sum_{i:(i,j)\in\mathcal{A}} u_{ij}^k + q_j x_{ij}^k = \sum_{i:(j,i)\in\mathcal{A}} u_{ji}^k \qquad \forall j \in \mathcal{N} \setminus \mathcal{O}, k \in \mathcal{K} \tag{2.16}$$

Constraints (2.15) are linking constraints between variables x_{ij}^k and u_{ij}^k; they impose that vehicle k cannot transport load from i to j if the vehicle is not traversing that arc $\left(x_{ij}^k = 0 \implies u_{ij}^k = 0\right)$. Constraints (2.16) state that the difference between the vehicle load when entering and leaving a customer must be equal to the demanded quantity, except for depot tasks for finishing and starting trips, if applicable. For the sake of generality, we assume that $q_i < 0$ if load is to be delivered at customer i and $q_i > 0$ if load is to be collected. These constraints, together with constraints (2.40), establish the capacity limits that must be imposed on each vehicle. When multiple trips are acknowledged, additional capacity constraints are typically required to ensure that load variables u_{ij}^k are reset before starting eventual new trips. In these situations, new constraints (2.17) and (2.18) are introduced.

$$\sum_{o:(i,o)\in\mathcal{A}} u_{io}^k = 0 \qquad \forall k \in \mathcal{K}, o \in \mathcal{O}^k \tag{2.17}$$

$$\sum_{j:(o,j)\in\mathcal{A}} u_{oj}^k = 0 \qquad \forall k \in \mathcal{K}, o \in \mathcal{O}^k \tag{2.18}$$

Constraints (2.17) establish that the total vehicle load is equal to zero when finishing a trip.

Constraints (2.18) establish that the total vehicle load is equal to zero when starting a new trip. Depending on the specific problem being tackled, at least one of these constraints is required. For routing problems whose purpose is to deliver load from the depot location to customers, constraints (2.17) must be applied. For routing problem whose purpose is to collect load from customers to the depot location, constraints (2.18) must be applied instead. If, instead, the problem consists in picking up and delivering load between tasks, similarly to a Pickup and Delivery Problem, then both constraints must be applied.

Operation constraints. The interdependencies triggered by operations within a route have two major types of constraints, which can be applied depending on the specific application.
Type 1 constraints. These constraints establish the sets of tasks that a route can perform immediately after a given task is performed. In other words, these constraints allow for the definition of the possible sequences of tasks (or arcs) that vehicles may perform (or traverse) in the course of their routes.
These interdependencies can be modelled by adding the following constraints and considering set $\mathcal{R}$ and $\mathcal{E}_i$, as well as parameters δ_{ij} and Δ_{ij}.

$$\sum_{l:(l,i)\in\mathcal{A}} x_{li}^k - \sum_{j\in\mathcal{E}_i} x_{ij}^k = 0 \qquad \forall i \in \mathcal{N}_0 : |\mathcal{E}_i| > 0, k \in \mathcal{K} \tag{2.19}$$

Constraints (2.19) establish that a task j from set $\mathcal{E}_i$ will need to be performed immediately after task i. In practice, and as a consequence of the effect of these constraints, these operation constraints provide a preprocessing of the transportation network. For each task i subject to these operation constraints, each potential subsequent task $j \notin \mathcal{E}_i$ will necessarily form an arc (i, j) that is infeasible, and, therefore, can be removed from $\mathcal{A}$.
Type 2 constraints. These constraints, on the other hand, have the purpose of imposing time offsets between tasks of an operation that is being performed by the same route. Therefore, these constraints will only be binding if the operation is performed in the same route.
These interdependencies can be modelled by adding the following constraints and considering set $\mathcal{R}$ and parameters δ_{ij} and Δ_{ij}.

$$w_i^k + \delta_{ij} \le w_j^k + T\left(1 - y_{ij}^{kk}\right) \qquad \forall (i, j) \in \mathcal{R}, k \in \mathcal{K} \tag{2.20}$$

$$w_i^k + \Delta_{ij} + T\left(1 - y_{ij}^{kk}\right) \ge w_j^k \qquad \forall (i, j) \in \mathcal{R}, k \in \mathcal{K} \tag{2.21}$$

Constraints (2.20) and (2.21) are the type 2 constraints that impose time offsets between tasks of a given operation (i, j). Specifically, constraints (2.20) set a minimum time offset between performing task i and performing task j. They ensure that task j can only start being performed δ_{ij} time units after starting to perform task i. Analogously, constraints (2.21) set a maximum time offset between performing task i and performing task j. They ensure that task j must start being performed up to Δ_{ij} time units after starting to perform task i.

Time window constraints. Time windows are easily modelled through the addition of the following constraints and by considering parameters a_i and b_i.

$$\sum_{i:(i,j)\in\mathcal{A}} a_j x_{ij}^k \leq w_j^k \leq \sum_{i:(i,j)\in\mathcal{A}} b_j x_{ij}^k \qquad \forall j \in \mathcal{N}, k \in \mathcal{K} \tag{2.22}$$

These constraints (2.22) establish lower and upper bounds to the arrival time of a vehicle to a given task, according to its desired time windows.

Route length constraints. Limiting the route length of a vehicle can either be performed through the total duration of the route or through the total travelled distance.
If parameter D^k refers to a route's total travelled distance, then the problem aspect is modelled as follows.

$$\sum_{(i,j)\in\mathcal{A}} d_{ij} x_{ij}^k \leq D^k \qquad \forall k \in \mathcal{K} \tag{2.23}$$

Equations (2.23) limit the total travelled distance of each route up to parameter D^k.
If, instead, parameter D^k refers to a route's total duration, then different constraints must be introduced.

$$w_{N+1}^k \leq D^k \qquad \forall k \in \mathcal{K} \tag{2.24}$$

Equations (2.24) limit the total duration of each route up to parameter D^k.
For routing problems acknowledging multiple trips, besides limiting the duration of the complete route, it is also common to limit the duration of each trip. In these cases, a new parameter D'_k will designate the maximum duration of each trip in route $k \in \mathcal{K}$. For each vehicle $k \in \mathcal{K}$, we consider set $\mathcal{O}^k = \{o_1, ..., o_O\}$, of cardinality O, and assume that these depot intermediate tasks must be performed in a pre-established order. With this is mind, let auxiliary set $\mathcal{O}'_k = \{(o_1, o_2), (o_2, o_3), ..., (o_{O-1}, o_O)\}$ be the ordered pairs of intermediate tasks that establish this order. Under these conditions, the following constraints apply.

$$\sum_{i:(i,o)\in\mathcal{A}} x_{io}^k \geq \sum_{i:(i,o')\in\mathcal{A}} x_{io'}^k \qquad \forall k \in \mathcal{K}, (o,o') \in \mathcal{O}'_k \tag{2.25}$$

$$w_{o_1}^k \leq w_0^k + D'_k \qquad \forall k \in \mathcal{K} \tag{2.26}$$

$$w_{o'}^k \leq w_o^k + D'_k \qquad \forall k \in \mathcal{K}, (o,o') \in \mathcal{O}'_k \tag{2.27}$$

$$w_{N+1}^k \leq w_{o_{O-n}}^k + nD'_k \qquad \forall k \in \mathcal{K}, n = 0, ..., O-1 \tag{2.28}$$

Constraints (2.25) establish that, if multiple trips are performed in a route, the depot intermediate tasks must be performed in a pre-established order. Constraints (2.26)–(2.28) limit the total duration of each trip up to parameter D'_k. To that effect, constraints (2.26) limit the duration of the first trip and constraints (2.27) limit the duration of the trips that follow.

Finally, constraints (2.28) limit the duration of the last trip that is performed.

2.5.6 Synchronisation Aspects

Operation synchronisation. For modelling purposes, it is relevant to distinguish two types of synchronisation constraints: lower-bounding (LB) and upper-bounding (UB) constraints. As their names imply, LB constraints set a lower bound on the start time of the second task. This lower bound is calculated from the time that the first task starts being executed. On the other hand, UB constraints set an upper bound on the start time of the second task.

By applying these synchronisation constraints, as well as different combinations of them, we are able to model a variety of synchronisation requirements, identical to the ones present in Dohn et al. [2011].

Figure 2.8 illustrates the possible combinations of synchronisation constraints in a Gantt chart-like form. A given operation (i, j) requires that tasks i and j be performed at their corresponding locations. We assume that they will be visited by two distinct vehicles k and k', respectively. In this figure, it is possible to visualise the inadmissible service time allocations (marked in dashed form) that each of the synchronisation types entails. For example, in Figure 2.8a, the diagram shows that task i can only be delayed up to a certain time without also having to delay task j; analogously, task j can only be moved up to a certain time without also having to move up task i.

For the sake of simplicity, we will use the following notation $i \preceq j$ to indicate that task i is visited before task j (or simultaneously).

Minimum difference (Figure 2.8a) is the result of applying a lower-bounding synchronisation constraint between tasks i and j. It is a synchronisation type that forces $i \preceq j$ and can be found in several real-world applications, such as in large items transportation, where delivery and installation vehicles must be synchronised such that the delivery of the merchandise must occur before the installing team arriving [e.g., Hojabri et al., 2018]. This happens because the time offset $\lambda_{ij} \geq 0$ is implemented with respect to the arrival time of vehicle k at task i, w_i^k. A dependence relationship is established between i and j, where the visit to task i can be moved up as much as desired, and the visit to task j can be delayed as much as necessary.

Maximum difference (Figure 2.8b) is the result of applying an upper-bounding constraint between tasks i and j. Unlike the minimum difference type, this synchronisation type does not force a given order of visits. One of the most representative examples of this synchronisation type concerns the pickup and delivery of perishable items involving multiple vehicles, where the time lapse between having the pickup vehicle picking up the merchandise and having the delivery vehicle deliver it must be constrained [e.g., Anaya-Arenas et al., 2019]. This is due to the fact that in this synchronisation type we are only concerned with maintaining a maximum offset between task arrival times. In the case illustrated in Figure 2.8b, one can observe that tasks i and j can only be moved up or delayed up to a certain threshold $\mu_{ij} \geq 0$, respectively. However, tasks i and j can be delayed or moved up without restrictions, respectively, meaning that an order of visits is not forced. In conclusion, this synchronisation type will only be binding if $i \preceq j$.

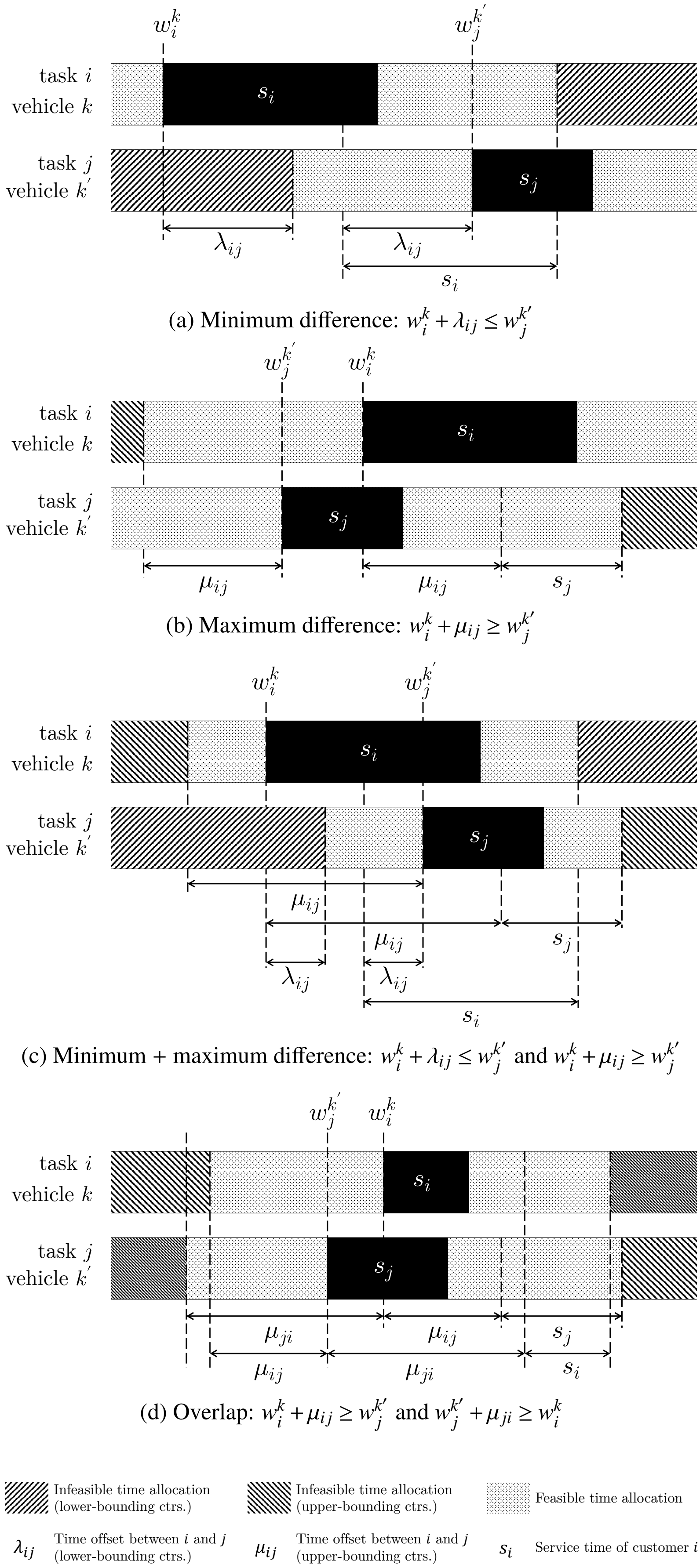


Figure 2.8: Types of synchronisation

The synchronisation types that follow consist of different possible combinations of the constraints presented above. Figure 2.8c illustrates a case of minimum + maximum difference, which combines lower-bounding and upper-bounding constraints. Unlike the previous cases, arrival times are now constrained within a closed interval due to combining both synchronisation types. We continue to force a given order of visits and we require that services not be performed too far apart. A special case of this synchonisation type is exact synchronisation, where $\lambda_{ij} = \mu_{ij}$. For this special case, the time offset between the arrivals of i and j must be exact. One may find this synchronisation type in several applications in the literature, such as home health care routing [e.g., Mankowska et al., 2013], where two or more staff members may be required to be present simultaneously at a patient's location, in which case $\lambda_{ij} = \mu_{ij} = 0$.
Finally, Figure 2.8d illustrates a case of overlap, which results from the combination of two upper-bounding constraints: one between i and j and another one between j and i. Overlap is relatively similar to the minimum + maximum difference type; however, there is one major difference. Although the arrival times at tasks are also constrained within a closed interval, in overlap, we do not force a given order of visits.
We present a summary of these synchronisation types in Table 2.4 with regard to the main aspects of each operation synchronisation type.
It should be noted that, although we have used the arrival time as the temporal variable to be synchronised, this could also be applied *mutatis mutandis* to other variables (e.g., a task's completion time) without significantly changing the theoretical aspects of each synchronisation type, such as task end times.

Table 2.4: Caracteristics of different types of synchronisation

Synchronisation type	**Constraints**			**Order of visits**	**Time lapse between visits**	
	LB (i,j)	UB (i,j)	UB (j,i)		Minimum	Maximum
Minimum difference	•	–	–	$i \leq j$	$\lambda_{ij} \geq 0$	–
Maximum difference	–	•	–	–	–	$\mu_{ij} \geq 0$ (if $i \leq j$)
Min. + max. difference	•	•	–	$i \leq j$	$\lambda_{ij} \geq 0$	$\mu_{ij} \geq 0$
Overlap	–	•	•	–	–	$\mu_{ij} \geq 0$ (if $i \leq j$) $\mu_{ji} \geq 0$ (if $j \leq i$)
Exact synchronisation	•	•	–	$i \leq j$	$\lambda_{ij} \geq 0, \lambda_{ij} = \mu_{ij}$	$\mu_{ij} \geq 0, \mu_{ij} = \lambda_{ij}$

For modelling operation synchronisation requirements, we consider sets $\mathcal{P}$ and $\mathcal{H}$, which will contain the operations subject to lower-bounding and upper-bounding constraints, respectively ($\mathcal{P} \subseteq \mathcal{R}, \mathcal{H} \subseteq \mathcal{R}$).
For operations (i,j) subject to both lower-bounding and upper-bounding constraints, these pairs are included both in sets $\mathcal{P}$ and $\mathcal{H}$. In the case of the overlap type, both operations (i,j) and (j,i) are added to set $\mathcal{H}$.
The time offset between the start times of task pairs (i,j) is represented by λ_{ij} for lower-bounding constraints and μ_{ij} for upper-bounding constraints.
The baseline VRP formulation is now updated with the following constraints.

$$w_i^k + \lambda_{ij} \leq w_j^{k'} + T\left(1 - y_{ij}^{kk'}\right) \qquad \forall (i,j) \in \mathcal{P}, k, k' \in \mathcal{K} : k \neq k' \tag{2.29}$$

$$w_i^k + \mu_{ij} + T\left(1 - y_{ij}^{kk'}\right) \geq w_j^{k'} \qquad \forall (i,j) \in \mathcal{H}, k, k' \in \mathcal{K} : k \neq k' \tag{2.30}$$

Constraints (2.29) ensure that, for a given synchronised operation $(i, j) \in \mathcal{P}$, the start time of task j can only be performed λ_{ij} time units after the start time of task i. On the other hand, constraints (2.30) state that, for a given synchronised operation $(i, j) \in \mathcal{H}$, if $i \preceq j$, then task j must start being performed up to μ_{ij} time units after i begins to be performed.

Movement synchronisation. Movement synchronisation is achieved by considering parameter $z_{ij}^{i'j'}$, which translates the pairs of operations to be synchronised, as well as parameter p_{ij}, which establishes whether a given operation is passive or not. The VRP formulation is then updated with the synchronisation constraints that follow.

$$\sum_{k \in \mathcal{K}} x_{ij}^k \leq \sum_{k \in \mathcal{K}} \sum_{(i',j'):z_{ij}^{i'j'}=1} x_{i'j'}^k \qquad \forall (i,j) \in \mathcal{R} : p_{ij} = 1 \tag{2.31}$$

Constraints (2.31) state that a passive operation (i, j) can only be performed if there is also a corresponding active operation (i', j') being performed.

2.5.7 Global Aspects

Task constraints. Task constraints constitute a fundamental component of any routing problem; therefore, this problem aspect is intrinsically present in the baseline formulation, more specifically in constraints (2.3). For the sake of simplicity, in this formulation, we assume that there is only one type of task constraint among all tasks. However, a VRP with multiple operational constraints may require differentiated task constraints for different types of tasks, in which case different types of constraints will apply for each task. In our mathematical formulation, we assume that each task must be performed exactly once, and therefore the number of total visits to a given location is known. Depending on the specific VRP variant being modelled, these task constraints may require adjustments for specific subsets of $\mathcal{N}$, namely for tasks that are not mandatory. This may be the case for multi-trip VRPs, for example, where it may be necessary to account for intermediate optional tasks, located at the depot, in order to have a vehicle performing multiple legs from the depot in the course of its route.
Considering auxiliary subset $\mathcal{N}' \subset \mathcal{N}$ containing all optional tasks, constraints (2.3) are reformulated accordingly for these cases.

$$\sum_{k \in \mathcal{K}} \sum_{i:(i,j) \in \mathcal{A}} x_{ij}^k \leq 1 \qquad \forall j \in \mathcal{N}' \tag{2.32}$$

Constraints (2.32) present alternative task constraints, establishing that each optional task can be performed at most once, although it is not required to be performed.

Operation constraints. The purpose of operation constraints is to ensure that operations are performed under their correct conditions. These conditions may consist in ensuring that a given operation is mandatory or optional or even establishing that a given operation must/can be performed by the same or different vehicles. Considering auxiliary subsets $\mathcal{R}'$ to designate all optional operations, $\mathcal{R}''$ to contain all operations that must be performed by different vehicles and $\mathcal{R}'''$ to represent all operations that must be performed by the same vehicle, the following operation constraints can be introduced.

$$\sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}} y_{ij}^{kk'} \leq 1 \qquad \forall (i,j) \in \mathcal{R}' \tag{2.33}$$

$$\sum_{k \in \mathcal{K}} y_{ij}^{kk} = 0 \qquad \forall (i,j) \in \mathcal{R}'' \tag{2.34}$$

$$\sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}: k \neq k'} y_{ij}^{kk'} = 0 \qquad \forall (i,j) \in \mathcal{R}''' \tag{2.35}$$

Constraints (2.33) establish that, for a given operation, it may or not be performed. Constraints (2.34) are applied to an operation that cannot be performed by the same vehicle: in these situations, the operation must forcefully be performed by different vehicles. Constraints (2.35) are applied to an operation that cannot be performed by different vehicles: in these situations, the operation must forcefully be performed by the same vehicle.

Resource constraints. The implementation of global resource constraints in a mathematical formulation is highly dependent on the nature of the resource being controlled. As an example, we will present the implementation of demand constraints for a VRP with Split Deliveries. In this problem variant, locations typically have more than one delivery task, and it is necessary to ensure that the total load that is left at each task is equal to the location's demand.

The nature of the split deliveries problem aspect is basically the same as any other type of resource constraints: we have a bounded/limited resource (in this case, the demand of a location) that requires vehicles to coordinate the amount of load between themselves, so that the global constraint is satisfied (in this case, that the total load from vehicles equals the location's demand), thus triggering inter-vehicle competition and trade-offs.

For modelling the demand requirements, we assume that each task can be performed at most once, although it is not required to be performed. Therefore, constraints (2.3) are replaced by constraints (2.32).

Parameter q_i becomes associated with locations instead of tasks. Therefore, parameter q_l will now represent the total demand to be satisfied by location $l \in \mathcal{L}$. Analogously to what was previously stated for q_i, we assume that $q_l < 0$ if load is to be delivered at l and $q_l > 0$ if load is to be picked up at l.

Constraints (2.16) are removed, and new global constraints are introduced.

$$\sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{N}:\psi(j)=l}\sum_{i:(j,i)\in\mathcal{A}} u_{ji}^k - \sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{N}:\psi(j)=l}\sum_{i:(i,j)\in\mathcal{A}} u_{ij}^k = q_l \qquad \forall l\in\mathcal{L} \tag{2.36}$$

Constraints (2.36) reflect the demand constraints for each location. They impose that the difference between the sum of vehicle loads entering tasks of location l (first summand) and the sum of vehicle loads exiting these same tasks (second summand) must be equal to the total demand of location l.

2.5.8 Integration Aspects

The development of a modelling framework for the VRP with integration aspects is out of the scope of this paper. This is due to the fact that mathematical formulations with integration aspects are case-specific, depending on the process being integrated. This precludes the possibility of having a general formulation with these aspects. For examples on integrated VRPs we refer the reader to Cordeau et al. [2015] for inventory routing, Adulyasak et al. [2015], Neves-Moreira et al. [2019] for production routing and Schneider and Drexl [2017] for location routing.

2.5.9 Domain and nature of the decision variables

The domain and nature of the decision variables used throughout the model are now presented.

$$x_{ij}^k \in \{0,1\} \qquad \forall (i,j)\in\mathcal{A}, k\in\mathcal{K} \tag{2.37}$$

$$y_{ij}^{kk'} \in \{0,1\} \qquad \forall (i,j)\in\mathcal{R}, k,k'\in\mathcal{K} \tag{2.38}$$

$$0 \le w_i^k \le T \qquad \forall i\in\mathcal{N}_0, k\in\mathcal{K} \tag{2.39}$$

$$0 \le u_{ij}^k \le Q^k \qquad \forall (i,j)\in\mathcal{A}, k\in\mathcal{K} \tag{2.40}$$

Variables x_{ij}^k and $y_{ij}^{kk'}$ are defined as binary variables through constraints (2.37) and (2.38), respectively. Constraints (2.39) define variables w_i^k as continuous, whose values cannot exceed the established planning horizon. Analogously, constraints (2.40) also define variables u_{ij}^k as continuous, whose values cannot exceed the maximum capacity of the vehicle associated with each variable.

2.6. Instantiation and validation of the modelling framework

This section outlines the main adaptations to the proposed modelling framework that are required to successfully model several VRP variants.

2.6.1 VRP with Time Windows

The VRP with Time Windows (VRPTW) consists in a Capacitated VRP (CVRP) where each customer is visited exactly once, and visits to customers must occur between given time limits [Savelsbergh, 1992]. For each of the customers to be visited, there will be a single task being performed, and therefore $|\mathcal{N}| = |\mathcal{L}|$. Modelling the VRPTW with the proposed modelling framework is trivial since it builds on the instantiation of a CVRP. To instantiate a CVRP with the proposed modelling framework, it is necessary to acknowledge the fundamental routing constraints (2.3)–(2.14) along with capacity constraints (2.16) and constraints (2.37)–(2.40), acknowledging the nature and domain of the decision variables. The VRPTW is then instantiated by adding time window constraints (2.22) for each customer. The demand of task i, q_i, will correspond to the total demand of its corresponding location/customer, $\psi(i)$. In this case, since the routing problem only includes delivery tasks, $q_i < 0$.

2.6.2 Home Health Care Routing and Scheduling Problem

The Home Health Care Routing and Scheduling Problem (HHCRSP) is a routing problem whose purpose is to obtain a set of routes for home health care staff in order to visit a set of patients that require certain services to be executed at their locations [e.g., Mankowska et al., 2013]. The problem needs to acknowledge different staff qualifications for different services and by having some synchronisation requirements at patients. Furthermore, certain services at customers must be synchronised, i.e., they must be performed simultaneously by more than one staff member or within a given temporal offset.

The HHCRSP can easily be applied to the proposed formulation by considering the following adaptations:

- *Staff members and qualifications*: in this case, a staff member corresponds to a given vehicle, which in turn corresponds to a route. Each staff member is only able to perform certain tasks according to their qualifications. These qualifications are easily translated into binary parameter r_i^k, which will state if a given staff member k is able to perform task/service i.
- *Home Health Care services*: in this particular problem, each service corresponds to a task that needs to be satisfied by a staff member at the patient's corresponding location $\psi(i)$. The HHCRSP also assumes there are some services that must be performed by multiple staff members. In these specific cases, synchronised services are split into two or more tasks (depending on the number of staff members required), being these tasks later intertwined with operation synchronisation constraints.
- *Simultaneous services*: a simultaneous service is characterised by having staff members starting it exactly at the same time. The proposed modelling framework is able to handle this situation graciously. For an operation (i, j) corresponding to the tasks of a simultaneous service, we are able to guarantee that $w_i^k = w_j^k$ by applying exact synchronisation (constraints (2.29) and (2.30)) to (i, j), with $\lambda_{ij} = \mu_{ij} = 0$.
- *Services with precedence*: a service that must be performed before another is easily modelled through lower-bounded operation synchronisation for a given operation

(i, j), with $\lambda_{ij} > 0$. In specific cases, it may be required to ensure that service j cannot be performed until service i is finished (i.e. no service overlap), which can be achieved by setting $\lambda_{ij} = s_i$.

2.6.3 Pickup and Delivery Problem with Transfers

The Pickup and Delivery Problem with Transfers (PDPT) is a generalisation of the standard Pickup and Delivery Problem (PDP), which, in turn, is a generalisation of the CVRP.
The PDP acknowledges two main types of locations: pickup and delivery customers, which are previously paired [e.g., Parragh et al., 2008]. The main additional requirements of a PDP compared to a CVRP consist in the fact that pickup and delivery of a given request must be performed by the same vehicle and that pickup must be performed before delivery. The PDP with Transfers extends upon the assumptions of the PDP by allowing pickup and delivery tasks to be performed by different vehicles [e.g., Masson et al., 2013b]. To that effect, transfer locations exist where picked-up load can be transferred from one vehicle to another.
In these circumstances, the PDPT is modelled by considering the following aspects to the instantiation of the CVRP, described previously:

- *Pickup and delivery tasks*: each pickup and delivery request, composed of pickup task i and delivery task j, each one of them located at its corresponding customers, is considered a mandatory operation $(i, j) \in \mathcal{R}$, which must be performed by either the same or different vehicles.
- *Transfer tasks*: for each transfer location and pickup and delivery request (i, j) present in the problem, two additional tasks i' and j' must be considered, where task i' represents the drop-off of the load of (i, j) at the transfer location and task j' represents the pickup of that transfered load.
- *Local constraints*: besides the traditional capacity constraints, it is also necessary to ensure that, if a pickup and delivery request is performed by only one vehicle (i.e., the request is fulfilled without transfer), the pickup task i must be performed before its corresponding delivery task j. Furthermore, it is also necessary to ensure temporal precedence for operations (i, i') and (j', j), if they are performed. For the request (i, j), this is achieved by applying local operation constraints (2.20) to each request (i, j), with $\delta_{ij} = 0$, or, alternatively, $\delta_{ij} = s_i$, if tighter constraints are preferred. For operations (i, i') and (j', j), local operation constraints (2.20) are also applied with $\delta_{ii'} = 0$ and $\delta_{j'j} = 0$, respectively, or $\delta_{ii'} = s_i$ and $\delta_{j'j} = s_{j'}$, if tighter constraints are preferred.
- *Optional tasks*: all tasks present at transfer locations are not mandatory, and therefore, global task constraints (2.32) are applied to these cases instead of standard constraints (2.3).
- *Operations concerning transfers*: taking into account the problem requirements, pairs (i, i') and (j', j) constitute optional operations, which, if performed, must be performed by the same vehicles. Therefore, in these cases, global operation constraints (2.33) and (2.35) apply. Furthermore, pairs (i', j') also constitute optional operations that, if performed, must be performed by different vehicles. In these in-

stances, global operation constraints (2.33) and (2.34) apply.

- *Synchronising transfer*: for operations (i', j'), being performed in a transfer, it is necessary to ensure that a vehicle visiting task j' is not being performed before task i' is performed by a different vehicle. Taking this requirement into account, operations (i', j') are subject to operation synchronisation, ensuring a minimum difference between the arrival times of both vehicles. Therefore, lower-bounding synchronisation constraints (2.29) are applied to operations (i', j'), with $\lambda_{i'j'} \geq 0$ defining the minimal difference between the arrival times.

2.6.4 Truck and Trailer Routing Problem

The Truck and Trailer Routing Problem (TTRP) is a generalisation of the VRP, which is characterised for acknowledging two different types of vehicles – trucks (active vehicles) and trailers (passive vehicles) [e.g., Chao, 2002]. Trailers cannot move between locations without a truck transporting them. In this problem, there are site-dependency constraints related to trailers only visiting certain customers, which splits the customer set into truck-only customers and vehicle customers, which allow trailers in their locations. This additional practical constraint leads to the need for a truck leaving the trailer at a vehicle customer and perform a sub-tour to other customers. In sum, trucks may perform three types of routes: routes with no trailer attached to it, routes with a trailer coupled to it at all times – in which case it can only visit vehicle customers –, and routes where trailers are temporarily left at a vehicle customer to serve truck-only customers.

The TTRP can also be modelled using the proposed mathematical formulation by performing some adjustments to it as follows:

- *Tasks involved in the problem*: because of the additional complexity provided by the TTRP, vehicle customer locations will have more than one task in order to account for the arrival of multiple vehicles to locations, as well as the multiple visits being performed by the same vehicle. With this in mind, there will be only one task for each truck-only location, which translates the delivery of the load required by the customer. As for vehicle customer locations, there will be four major types of tasks: (i) tasks representing the arrival of the truck ("truck arrival tasks"); (ii) tasks representing the (eventual) arrival of a trailer coupled to the truck ("trailer arrival tasks"); (iii) tasks representing the return of the truck to the customer after decoupling from the trailer and performing a truck-only subtour ("truck return tasks"); (iv) tasks for having the trailer recouple to the truck ("trailer coupling tasks"). Depending on the type of task, it should be performed by either a truck or a trailer. Therefore, parameter r_i^k should also be adjusted to 0 for incompatible task and vehicle combinations.
- *Optional tasks*: for tasks at truck-only locations, all tasks must be considered mandatory. However, in vehicle customer locations, due to the multitude of situations that can be verified, only truck arrival tasks are mandatory. Because trailers may not be required to visit vehicle customers if the truck is able to satisfy the customer's demand, trailer arrival tasks are optional, and therefore do not require being satisfied. Consequently, because truck return tasks and trailer coupling tasks only make sense to be performed if trailer delivery tasks are also performed, these too are considered

optional. This requirement translates into the mathematical formulation by applying the global task constraints presented in constraints (2.32) for these specific types of tasks. Truck delivery tasks, on the other hand, will continue to be guaranteed by constraints (2.3).

- *Demand satisfaction*: in the TTRP, the demand of vehicle customers can be satisfied by any of the vehicles that arrive at its location – i.e., the truck or the trailer. Therefore, since the allocation of demand from each of the vehicles becomes unknown *a priori*, the problem requires the addition of new decision variables. This problem requirement is successfully modelled by considering global resource constraints (2.36) along with the new underlying decision variables.
- *Task precedences within the same route*: due to the problem's intrinsic requirements, there are some route sequences in the TTRP that cannot happen. For example, a truck delivery task cannot be performed without either performing a trailer delivery task or another truck delivery task immediately before. Another example consists in trailer coupling tasks, which must be immediately followed by a trailer delivery task. These task precedence requirements are achieved through local operation constraints. To successfully acknowledge this problem requirement, we add all the possible task sequences (i, j) that may occur within a route to set $\mathcal{R}$ and apply constraints (2.19). Alternatively, preprocessing procedures can be devised so that the problem's graph incorporates only feasible task sequences.
- *Truck and trailer simultaneous movement*: modelling the simultaneous movement between trucks and trailers is relatively painless by using the operation and movement synchronisation constraints previously presented. Considering a passive operation (i, j), where $p_{ij} = 1$, which corresponds to a possible task sequence/arc being performed by a trailer, and also considering all active operations (i', j') that can be synchronised with operation (i, j), where $z_{ij}^{i'j'} = 1$, movement synchronisation is therefore applied for every task sequence combination (i, j) of a trailer where it needs to be transported by a truck, i.e., where $\psi(i) \neq \psi(j)$. Constraints (2.31) are added considering these mathematical abstractions. Additionally, it is also necessary to ensure that these task sequences are performed at the same time, which is why it is also necessary to add operation synchronisation constraints between these task sequences. For modelling this simultaneous movement, it is only necessary to ensure that the arrival time of the trailer at task j is exactly equal to the arrival time of the truck at either j', which corresponds to adding two operations (j, j') and (j, j'') to sets $\mathcal{P}$ and $\mathcal{H}$, with $\lambda_{jj'} = \lambda_{jj''} = \mu_{jj'} = \mu_{jj''} = 0$ and adding constraints (2.29) and (2.30).
- *Global operation constraints*: one of the requirements of the TTRP is that if a truck leaves a trailer at a vehicle customer to perform a truck-only subtour, then it must be the same truck performing the delivery at that vehicle customer that must also recouple the trailer that was left there. This problem requirement is accomplished by applying global operation constraints (2.35) to each pair of truck delivery tasks and truck return tasks within the same vehicle customer location.

2.7. Conclusions

This paper has introduced a classification schema and general modelling framework for VRPs with multiple synchronisation constraints, a topic under development in the literature.
The proposed classification schema allowed to classify different problem aspects in terms of the direct interdependency sizes they trigger in the routing problem: whether it is within a single route, multiple routes, among the entire routing plan or between different processes of the supply chain.
The proposed modelling framework intends to be as general as possible in order for it to be a "one-size-fits-all" formulation for routing problems with synchronisation constraints. The used mathematical notation is able to decompose each of the problem aspects of the proposed conceptual schema and propose a modelling approach for each of these problem aspects, thus obtaining a highly modular formulation, which can be leveraged to model different problem aspects with interaction with synchronisation aspects.
As it would be expected, such a general mathematical model yields some disadvantages. The first one consists in an increased initial effort in instantiating the formulation to a given problem variant or application. In fact, due to the general nature of the model, especially in questions of synchronisation, it may be necessary to perform several enumerations of tasks and operations in order to construct sets for the model. To mitigate this effort, it may be worthwhile to adapt the formulation and consider problem-specific sets and other modelling abstractions. The second disadvantage consists in the potential loss of model scalability and computational efficiency. Nevertheless, it should be noted that, depending on the problem variant and application being modelled, several simplifications can be applied to the formulation that can significantly enhance its scalability. One of them consists in network preprocessing procedures, which can eliminate inconsistent routing alternatives *a priori* based on the data obtained from instances and the interdependence requirements. For example, the presence of precedence aspects in synchronisation can potentiate the elimination of routing decisions that contradict said precedence requirements. Even different problem aspects may be combined to allow obtaining tighter models, e.g., time windows requirements with synchronisation requirements.
The performed literature review demonstrated that synchronisation is becoming an ever more interesting option for solving real-world transportation applications, having the potential to increase flexibility and reduce sources of inefficiency in transportation processes. Routing of autonomous vehicles has been one of the most prominent applications of synchronisation in recent years. Routing problems focused on the synchronisation of schedules are also fairly popular, as well as routing problems with movement synchronisation requirements, such as the joint routing of passive and active vehicles. Although there has been an effort in the development of solution methods for VRPs with synchronisation, it is the opinion of the authors that additional work must still be done to overcome the underlying complexity and combinatorial nature of these routing problems. The development of more efficient and effective solution methods will also be the key for having more routing problems that address sources of uncertainty, as well as new problem applications that resort to dynamic planning. It is the expectation of the authors that the research outputs of

this paper can help towards these goals.

Acknowledgments

The first author acknowledges the support of the Ph.D. Grant SFRH/BD/136314/2018, awarded by the Portuguese Science and Technology Foundation (FCT) and financed by the European Social Fund (ESF) and by National Funds of the Portuguese Ministry of Science, Technology and Higher Education (MCTES) through the Human Capital Operational Programme (POCH).

Bibliography

Yossiri Adulyasak, Jean-François Cordeau, and Raf Jans. The production routing problem: A review of formulations and solution algorithms. *Computers & Operations Research*, 55:141–152, 2015. doi: 10.1016/j.cor.2014.01.011.

Sohaib Afifi, Duc-Cuong Dang, and Aziz Moukrim. Heuristic solutions for the vehicle routing problem with time windows and synchronized visits. *Optimization Letters*, 10 (3):511–525, 2015. ISSN 1862-4472. doi: 10.1007/s11590-015-0878-3.

N Agatz, P Bouman, and M Schmidt. Optimization approaches for the traveling salesman problem with drone. *Transportation Science*, 52(4):965–981, 2018. doi: 10.1287/trsc.2017.0791.

S.R. Ait Haddadene, N. Labadie, and C. Prodhon. Bicriteria vehicle routing problem with preferences and timing constraints in home health care services. *Algorithms*, 12(8), 2019. ISSN 19994893. doi: 10.3390/a12080152.

Ousmane Ali, Jean-François Côté, and Leandro C. Coelho. Models and algorithms for the delivery and installation routing problem. *European Journal of Operational Research*, 291(1):162–177, May 2021. doi: 10.1016/j.ejor.2020.09.011.

H An, Y.-J. Byon, and C.-S. Cho. Economic and environmental evaluation of a brick delivery system based on multi-trip vehicle loader routing problem for small construction sites. *Sustainability (Switzerland)*, 10(5), 2018. doi: 10.3390/su10051427.

Ana María Anaya-Arenas, Caroline Prodhon, Jacques Renaud, and Angel Ruiz. An iterated local search for the biomedical sample transportation problem with multiple and interdependent pickups. *Journal of the Operational Research Society*, 72(2):367–382, December 2019. doi: 10.1080/01605682.2019.1657369.

A Anderluh, V C Hemmelmayr, and P C Nolz. Synchronizing vans and cargo bikes in a city distribution network. *Central European Journal of Operations Research*, 25(2): 345–376, 2017. doi: 10.1007/s10100-016-0441-z.

A. Anderluh, R. Larsen, V.C. Hemmelmayr, and P.C. Nolz. Impact of travel time uncertainties on the solution cost of a two-echelon vehicle routing problem with synchronization. *Flexible Services and Manufacturing Journal*, 2019a. ISSN 19366582. doi: 10.1007/s10696-019-09351-w.

A. Anderluh, P.C. Nolz, V.C. Hemmelmayr, and T.G. Crainic. Multi-objective optimization of a two-echelon vehicle routing problem with vehicle synchronization and 'grey zone' customers arising in urban logistics. *European Journal of Operational Research*, 2019b. ISSN 03772217. doi: 10.1016/j.ejor.2019.07.049.

C. Archetti and M. G. Speranza. Vehicle routing problems with split deliveries. *International Transactions in Operational Research*, 19(1-2):3–22, 2012. doi: 10.1111/j.1475-3995.2011.00811.x.

C Archetti, D Feillet, A Hertz, and M G Speranza. The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(6):831–842, 2009. doi: 10.1057/palgrave.jors.2602603.

Rikard Bakkehaug, Jørgen G. Rakke, Kjetil Fagerholt, and Gilbert Laporte. An adaptive large neighborhood search heuristic for fleet deployment problems with voyage separation requirements. *Transportation Research Part C: Emerging Technologies*, 70:129–141, 2016. doi: 10.1016/j.trc.2015.06.019.

Roberto Baldacci, Maria Battarra, and Daniele Vigo. Routing a heterogeneous fleet of vehicles. In *Operations Research/Computer Science Interfaces*, pages 3–27. Springer US, 2008. doi: 10.1007/978-0-387-77778-8_1.

Tolga Bektaş, Gilbert Laporte, and Daniele Vigo. Integrated vehicle routing problems. *Computers & Operations Research*, 55:126, 2015. doi: 10.1016/j.cor.2014.08.008.

Nicolas Bélanger, Guy Desaulniers, François Soumis, and Jacques Desrosiers. Periodic airline fleet assignment with time windows, spacing constraints, and time dependent revenues. *European Journal of Operational Research*, 175(3):1754–1766, 2006. doi: 10.1016/j.ejor.2004.04.051.

N. Bianchessi, M. Drexl, and S. Irnich. The split delivery vehicle routing problem with time windows and customer inconvenience constraints. *Transportation Science*, 53(4): 1067–1084, 2019. ISSN 00411655. doi: 10.1287/trsc.2018.0862.

Marie-Claude Bolduc, Jacques Renaud, and Benoit Montreuil. Synchronized routing of seasonal products through a production/distribution network. *Central European Journal of Operations Research*, 14(2):209–228, 2006. ISSN 1435-246X. doi: 10.1007/s10100-006-0169-2.

N Boysen, S Schwerdfeger, and F Weidinger. Scheduling last-mile deliveries with truck-based autonomous robots. *European Journal of Operational Research*, 271(3):1085–1099, 2018. doi: 10.1016/j.ejor.2018.05.058.

C. Brandstätter. A metaheuristic algorithm and structured analysis for the Line-haul Feeder Vehicle Routing Problem with Time Windows. *Central European Journal of Operations Research*, 2019. ISSN 1435246X. doi: 10.1007/s10100-019-00625-0.

C Brandstätter and M Reimann. The Line-haul Feeder Vehicle Routing Problem: Mathematical model formulation and heuristic approaches. *European Journal of Operational Research*, 270(1):157–170, 2018. doi: 10.1016/j.ejor.2018.03.014.

David Bredström and Mikael Rönnqvist. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, 191(1):19–31, 2008. doi: 10.1016/j.ejor.2007.07.033.

Paola Cappanera, Cristina Requejo, and Maria Grazia Scutellà. Temporal constraints and device management for the skill VRP: Mathematical model and lower bounding techniques. *Computers & Operations Research*, 124:105054, December 2020. doi: 10.1016/j.cor.2020.105054.

I-Ming Chao. A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29(1):33–51, 2002. doi: 10.1016/s0305-0548(00)00056-3.

Ching-Wu Chu. A heuristic algorithm for the truckload and less-than-truckload problem. *European Journal of Operational Research*, 165(3):657–667, 2005. doi: 10.1016/j.ejor.2003.08.067.

Sung Hoon Chung, Bhawesh Sah, and Jinkun Lee. Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions. *Computers & Operations Research*, 123:105004, November 2020. doi: 10.1016/j.cor.2020.105004.

M.-A. Coindreau, O. Gallay, and N. Zufferey. Vehicle routing with transportable resources: Using carpooling and walking for on-site services. *European Journal of Operational Research*, 279(3):996–1010, 2019. ISSN 03772217. doi: 10.1016/j.ejor.2019.06.039.

Marc-Antoine Coindreau, Olivier Gallay, and Nicolas Zufferey. Parcel delivery cost minimization with time window constraints using trucks and drones. *Networks*, February 2021. doi: 10.1002/net.22019.

Jean-François Cordeau, Demetrio Laganà, Roberto Musmanno, and Francesca Vocaturo. A decomposition-based heuristic for the multiple-product inventory-routing problem. *Computers & Operations Research*, 55:153–166, 2015. doi: 10.1016/j.cor.2014.06.007.

Iman Dayarian, Martin Savelsbergh, and John-Paul Clarke. Same-day delivery with drone resupply. *Transportation Science*, January 2020. doi: 10.1287/trsc.2019.0944.

Jérémy Decerle, Olivier Grunder, Amir Hajjam El Hassani, and Oussama Barakat. A memetic algorithm for a home health care routing and scheduling problem. *Operations Research for Health Care*, 16:59–71, 2018. doi: 10.1016/j.orhc.2018.01.004.

I. Deif and L. D. Bodin. Extension of the Clarke and Wright Algorithm for Solving the Vehicle Routing Problem with Backhauling. In *Proceedings of the Babson Conference on Software Uses in Transportation and Logistic Management*, pages 75–96, Babson Parl, 1984.

Nico Dellaert, Tom Van Woensel, Teodor Gabriel Crainic, and Fardin Dashty Saridarq. A multi-commodity two-echelon capacitated vehicle routing problem with time windows: Model formulations and solution approach. *Computers & Operations Research*, 127: 105154, March 2021. doi: 10.1016/j.cor.2020.105154.

M. Dell'Amico, R. Montemanni, and S. Novellani. Drone-assisted deliveries: New formulations for the flying sidekick traveling salesman problem. *Optimization Letters*, 2019. ISSN 18624472. doi: 10.1007/s11590-019-01492-z.

Ulrich Derigs, Markus Pullmann, and Ulrich Vogel. Truck and trailer routing – problems, heuristics and computational experience. *Computers & Operations Research*, 40(2): 536–546, February 2013. doi: 10.1016/j.cor.2012.08.007.

Anders Dohn, Esben Kolind, and Jens Clausen. The manpower allocation problem with time windows and job-teaming constraints: A branch-and-price approach. *Computers & Operations Research*, 36(4):1145–1157, 2009. doi: 10.1016/j.cor.2007.12.011.

Anders Dohn, Matias Sevel Rasmussen, and Jesper Larsen. The vehicle routing problem with time windows and temporal dependencies. *Networks*, 58(4):273–289, 2011. ISSN 1097-0037. doi: 10.1002/net.20472.

Bencomo Domínguez-Martín, Inmaculada Rodríguez-Martín, and Juan-José Salazar-González. The driver and vehicle routing problem. *Computers & Operations Research*, 92:56–64, 2018a. doi: 10.1016/j.cor.2017.12.010.

Bencomo Domínguez-Martín, Inmaculada Rodríguez-Martín, and Juan-José Salazar-González. A heuristic approach to the driver and vehicle routing problem. In *Lecture*

Notes in Computer Science, pages 295–305. Springer International Publishing, 2018b. doi: 10.1007/978-3-030-00898-7_19.

Hossein Hashemi Doulabi, Gilles Pesant, and Louis-Martin Rousseau. Vehicle routing problems with synchronized visits and stochastic travel and service times: Applications in healthcare. *Transportation Science*, 54(4):1053–1072, July 2020. doi: 10.1287/trsc.2019.0956.

M Drexl. Branch-and-cut algorithms for the vehicle routing problem with trailers and transshipments. *Networks*, 63(1):119–133, 2014. doi: 10.1002/net.21526.

Michael Drexl. Synchronization in vehicle routing—a survey of VRPs with multiple synchronization constraints. *Transportation Science*, 46(3):297–316, 2012. doi: 10.1287/trsc.1110.0400.

Mohsen Emadikhiav, David Bergman, and Robert Day. Consistent routing and scheduling with simultaneous pickups and deliveries. *Production and Operations Management*, 29 (8):1937–1955, June 2020. doi: 10.1111/poms.13200.

E Es Yurek and H C Ozmutlu. A decomposition-based iterative optimization algorithm for traveling salesman problem with drone. *Transportation Research Part C: Emerging Technologies*, 91:249–262, 2018. doi: 10.1016/j.trc.2018.04.009.

S Fedtke and N Boysen. Gantry crane and shuttle car scheduling in modern rail–rail transshipment yards. *OR Spectrum*, 39(2):473–503, 2017. doi: 10.1007/s00291-016-0461-z.

C Fikar, A A Juan, E Martinez, and P Hirsch. A discrete-event driven metaheuristic for dynamic home service routing with synchronised trip sharing. *European Journal of Industrial Engineering*, 10(3):323–340, 2016a. doi: 10.1504/EJIE.2016.076382.

Christian Fikar, Angel A. Juan, Enoc Martinez, and Patrick Hirsch. A discrete-event driven metaheuristic for dynamic home service routing with synchronised trip sharing. *European J. of Industrial Engineering*, 10(3):323, 2016b. doi: 10.1504/ejie.2016.076382.

Martin Fink, Guy Desaulniers, Markus Frey, Ferdinand Kiermaier, Rainer Kolisch, and François Soumis. Column generation for vehicle routing problems with multiple synchronization constraints. *European Journal of Operational Research*, 272(2):699–711, 2019. doi: 10.1016/j.ejor.2018.06.046.

S. Frifita and M. Masmoudi. VNS methods for home care routing and scheduling problem with temporal dependencies, and multiple structures and specialties. *International Transactions in Operational Research*, 27(1):291–313, 2020. ISSN 09696016. doi: 10.1111/itor.12604.

Aurélien Froger, Ola Jabali, Jorge E. Mendoza, and Gilbert Laporte. The electric vehicle routing problem with capacitated charging stations. *Transportation Science*, 56(2):460–482, March 2022. doi: 10.1287/trsc.2021.1111.

N. Furian, M. O'Sullivan, C. Walker, and S. Vössner. Evaluating the impact of optimization algorithms for patient transits dispatching using discrete event simulation. *Operations Research for Health Care*, 19:134–155, 2018. doi: 10.1016/j.orhc.2018.03.008.

V Ghilas, E Demir, and T V Woensel. A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. *Transportation Research Part B: Methodological*, 91:34–51, 2016a. doi: 10.1016/j.trb.2016.04.015.

Veaceslav Ghilas, Emrah Demir, and Tom Van Woensel. An adaptive large neighborhood search heuristic for the Pickup and Delivery Problem with Time Windows and Scheduled Lines. *Computers & Operations Research*, 72:12–30, 2016b. ISSN 0305-0548. doi: 10.1016/J.COR.2016.01.018.

A Goel and F Meisel. Workforce routing and scheduling for electricity network maintenance with downtime minimization. *European Journal of Operational Research*, 231 (1):210–228, 2013. doi: 10.1016/j.ejor.2013.05.021.

Marc Goetschalckx and Charlotte Jacobs-Blecha. The vehicle routing problem with backhauls. *European Journal of Operational Research*, 42(1):39–51, 1989. doi: 10.1016/0377-2217(89)90057-x.

Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *European Journal of Operational Research*, 254 (1):80–91, 2016. doi: 10.1016/j.ejor.2016.03.040.

Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking. *Computers & Operations Research*, 84:116–126, 2017. doi: 10.1016/j.cor.2017.03.004.

Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. The vehicle routing problem with cross-docking and resource constraints. *Journal of Heuristics*, 27(1-2):31–61, August 2019. doi: 10.1007/s10732-019-09423-y.

Axel Grimault, Nathalie Bostel, and Fabien Lehuédé. An adaptive large neighborhood search for the full truckload pickup and delivery problem with resource synchronization. *Computers & Operations Research*, 88:1–14, 2017. doi: 10.1016/j.cor.2017.06.012.

Timo Gschwind. A comparison of column-generation approaches to the synchronized pickup and delivery problem. *European Journal of Operational Research*, 247(1):60–71, 2015. doi: 10.1016/j.ejor.2015.06.017.

Timo Gschwind. Route feasibility testing and forward time slack for the synchronized pickup and delivery problem. *OR Spectrum*, 2018. doi: 10.1007/s00291-018-0544-0.

Minh Hoàng Hà, Tat Dat Nguyen, Thinh Nguyen Duy, Hoang Giang Pham, Thuy Do, and Louis-Martin Rousseau. A new constraint programming model and a linear programming-based adaptive large neighborhood search for the vehicle routing problem with synchronization constraints. *Computers & Operations Research*, 124:105085, December 2020. doi: 10.1016/j.cor.2020.105085.

Nizar El Hachemi, Michel Gendreau, and Louis-Martin Rousseau. A heuristic to solve the synchronized log-truck scheduling problem. *Computers & Operations Research*, 40(3): 666–673, 2013. doi: 10.1016/j.cor.2011.02.002.

Syrine Roufaida Ait Haddadene, Nacima Labadie, and Caroline Prodhon. A GRASP ILS for the vehicle routing problem with time windows, synchronization and precedence constraints. *Expert Systems with Applications*, 66:274–294, 2016. doi: 10.1016/j.eswa.2016.09.002.

S. Hanafi, R. Mansini, and R. Zanotti. The multi-visit team orienteering problem with precedence constraints. *European Journal of Operational Research*, 2019. ISSN 03772217. doi: 10.1016/j.ejor.2019.09.045.

H Hojabri, M Gendreau, J.-Y. Potvin, and L.-M. Rousseau. Large neighborhood search with constraint programming for a vehicle routing problem with synchronization constraints. *Computers and Operations Research*, 92:87–97, 2018. doi: 10.1016/j.cor.2017.11.011.

Z.-H. Hu and C Wei. Synchronizing vehicles for multi-vehicle and one-cargo transportation. *Computers and Industrial Engineering*, 119:36–49, 2018. doi: 10.1016/j.cie.2018.03.023.

Y.-H. Huang, C A Blazquez, S.-H. Huang, G Paredes-Belmar, and G Latorre-Nuñez. Solving the Feeder Vehicle Routing Problem using ant colony optimization. *Computers and Industrial Engineering*, 2018. doi: 10.1016/j.cie.2018.10.037.

Stefan Irnich, Paolo Toth, and Daniele Vigo. *Chapter 1: The Family of Vehicle Routing Problems*, chapter 1, pages 1–33. Society for Industrial and Applied Mathematics, 2014. doi: 10.1137/1.9781611973594.ch1.

P Kitjacharoenchai, M Ventresca, M Moshref-Javadi, S Lee, J M A Tanchoco, and P A Brunese. Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Computers and Industrial Engineering*, 129:14–30, 2019. doi: 10.1016/j.cie.2019.01.020.

Patchara Kitjacharoenchai and Seokcheon Lee. Vehicle routing problem with drones for last mile delivery. *Procedia Manufacturing*, 39:314–324, 2019. doi: 10.1016/j.promfg.2020.01.338.

Patchara Kitjacharoenchai, Byung-Cheol Min, and Seokcheon Lee. Two echelon vehicle routing problem with drones in last mile delivery. *International Journal of Production Economics*, 225:107598, July 2020. doi: 10.1016/j.ijpe.2019.107598.

E Lam and P V Hentenryck. A branch-and-price-and-check model for the vehicle routing problem with location congestion. *Constraints*, 21(3):394–412, 2016. doi: 10.1007/s10601-016-9241-2.

Gilbert Laporte, Martin Desrochers, and Yves Nobert. Two exact algorithms for the distance-constrained vehicle routing problem. *Networks*, 14(1):161–172, 1984. ISSN 00283045. doi: 10.1002/net.3230140113.

Gilbert Laporte, Hélène Mercure, and Yves Nobert. An exact algorithm for the asymmetrical capacitated vehicle routing problem. *Networks*, 16(1):33–46, 1986. doi: 10.1002/net.3230160104.

Hongqi Li, Haotian Wang, Jun Chen, and Ming Bai. Two-echelon vehicle routing problem with time windows and mobile satellites. *Transportation Research Part B: Methodological*, 138:179–201, August 2020a. doi: 10.1016/j.trb.2020.05.010.

Hongqi Li, Haotian Wang, Jun Chen, and Ming Bai. Two-echelon vehicle routing problem with satellite bi-synchronization. *European Journal of Operational Research*, 288(3): 775–793, February 2021. doi: 10.1016/j.ejor.2020.06.019.

Jiliu Li, Hu Qin, Roberto Baldacci, and Wenbin Zhu. Branch-and-price-and-cut for the synchronized vehicle routing problem with split delivery, proportional service time and multiple time windows. *Transportation Research Part E: Logistics and Transportation Review*, 140:101955, August 2020b. doi: 10.1016/j.tre.2020.101955.

R Liu, Y Tao, and X Xie. An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and synchronized visits. *Computers and Operations Research*, 101:250–262, 2019. doi: 10.1016/j.cor.2018.08.002.

E López-Aguilar, V Boyer, M A Salazar-Aguilar, and M Sbihi. Linear Formulations for the Vehicle Routing Problem with Synchronization Constraints. *Journal of Computer and Systems Sciences International*, 57(3):453–462, 2018. doi: 10.1134/S106423071803005X.

Z Luo, H Qin, W Zhu, and A Lim. Branch-and-price-and-cut for the manpower routing problem with synchronization constraints. *Naval Research Logistics*, 63(2):138–171, 2016. doi: 10.1002/nav.21683.

Dorota Slawa Mankowska, Frank Meisel, and Christian Bierwirth. The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, 17(1):15–30, 2013. doi: 10.1007/s10729-013-9243-1.

Guillaume Marques, Ruslan Sadykov, Jean-Christophe Deschamps, and Rémy Dupas. A branch-cut-and-price approach for the single-trip and multi-trip two-echelon vehicle routing problem with time windows. working paper or preprint, November 2020.

M A Masmoudi, M Hosny, E Demir, and N Cheikhrouhou. A study on the heterogeneous fleet of alternative fuel vehicles: Reducing CO2 emissions by means of biodiesel fuel. *Transportation Research Part D: Transport and Environment*, 63:137–155, 2018. doi: 10.1016/j.trd.2018.04.025.

R Masson, F Lehuédé, and O Péton. Efficient feasibility testing for request insertion in the pickup and delivery problem with transfers. *Operations Research Letters*, 41(3): 211–215, 2013a. doi: 10.1016/j.orl.2013.01.007.

R Masson, F Lehuédé, and O Péton. The dial-a-ride problem with transfers. *Computers and Operations Research*, 41(1):12–23, 2014. doi: 10.1016/j.cor.2013.07.020.

Renaud Masson, Fabien Lehuédé, and Olivier Péton. An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47 (3):344–355, August 2013b. doi: 10.1287/trsc.1120.0432.

Carl Axel Benjamin Medbøen, Magnus Bolstad Holm, Mohamed Kais Msakni, Kjetil Fagerholt, and Peter Schütz. Combining optimization and simulation for designing a robust short-sea feeder network. *Algorithms*, 13(11):304, November 2020. doi: 10.3390/a13110304.

J. Medina, M. Hewitt, F. Lehuédé, and O. Péton. Integrating long-haul and local transportation planning: The Service Network Design and Routing Problem. *EURO Journal on Transportation and Logistics*, 8(2):119–145, 2019. ISSN 21924376. doi: 10.1007/s13676-017-0114-7.

Frank Meisel and Herbert Kopfer. Synchronized routing of active and passive means of transport. *OR Spectrum*, 36(2):297–322, 2012. doi: 10.1007/s00291-012-0310-7.

S.M. Mirhedayatian, T.G. Crainic, M. Guajardo, and S.W. Wallace. A two-echelon location-routing problem with synchronisation. *Journal of the Operational Research Society*, 2019. ISSN 01605682. doi: 10.1080/01605682.2019.1650625.

Abood Mourad, Jakob Puchinger, and Tom Van Woensel. Integrating autonomous delivery service into a passenger transportation system. *International Journal of Production Research*, 59(7):2116–2139, 2021. doi: 10.1080/00207543.2020.1746850.

C C Murray and A G Chu. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54:86–109, 2015. doi: 10.1016/j.trc.2015.03.005.

Fábio Neves-Moreira, Bernardo Almada-Lobo, Jean-François Cordeau, Luís Guimarães, and Raf Jans. Solving a large multi-product production-routing problem with delivery time windows. *Omega*, 86:154–172, 2019. doi: 10.1016/j.omega.2018.07.006.

Phuong Khanh Nguyen, Teodor Gabriel Crainic, and Michel Toulouse. A tabu search for Time-dependent Multi-zone Multi-trip Vehicle Routing Problem with Time Windows. *European Journal of Operational Research*, 231(1):43–56, 2013. ISSN 0377-2217. doi: 10.1016/J.EJOR.2013.05.026.

Pamela C. Nolz, Nabil Absi, Diego Cattaruzza, and Dominique Feillet. Two-echelon distribution with a single capacitated city hub. *EURO Journal on Transportation and Logistics*, 9(3):100015, September 2020. doi: 10.1016/j.ejtl.2020.100015.

D C Paraskevopoulos, G Laporte, P P Repoussis, and C D Tarantilis. Resource constrained routing and scheduling: Review and research prospects. *European Journal of Operational Research*, 263(3):737–754, 2017. doi: 10.1016/j.ejor.2017.05.035.

Sophie N. Parragh and Jean-François Cordeau. Branch-and-price and adaptive large neighborhood search for the truck and trailer routing problem with time windows. *Computers & Operations Research*, 83:28–44, 2017. doi: 10.1016/j.cor.2017.01.020.

Sophie N. Parragh and Karl F. Doerner. Solving routing problems with pairwise synchronization constraints. *Central European Journal of Operations Research*, 26(2):443–464, 2018. doi: 10.1007/s10100-018-0520-4.

Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(2):81–117, 2008. doi: 10.1007/s11301-008-0036-4.

Z. Peng, Z. Al Chami, H. Manier, and M.-A. Manier. A hybrid particle swarm optimization for the selective pickup and delivery problem with transfers. *Engineering Applications of Artificial Intelligence*, 85:99–111, 2019. ISSN 09521976. doi: 10.1016/j.engappai.2019.06.006.

Hanne L. Petersen and Stefan Ropke. The pickup and delivery problem with cross-docking opportunity. In *Lecture Notes in Computer Science*, pages 101–113. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-24264-9_8.

David Pisinger and Stefan Ropke. A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8):2403–2435, August 2007. doi: 10.1016/j.cor.2005.09.012.

Mateusz Polnik, Annalisa Riccardi, and Kerem Akartunalı. A multistage optimisation algorithm for the large vehicle routing problem with time windows and synchronised visits. *Journal of the Operational Research Society*, pages 1–16, August 2020. doi: 10.1080/01605682.2020.1792365.

S.M. Pour, K. Marjani Rasmussen, J.H. Drake, and E.K. Burke. A constructive framework for the preventive signalling maintenance crew scheduling problem in the Danish railway system. *Journal of the Operational Research Society*, 70(11):1965–1982, 2019. ISSN 01605682. doi: 10.1080/01605682.2018.1507423.

Yuan Qu and Jonathan F. Bard. A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Computers and Operations Research*, 39(10):2439–2456, October 2012. doi: 10.1016/j.cor.2011.11.016.

A Rais, F Alvelos, and M S Carvalho. New mixed integer-programming model for the pickup-and-delivery problem with transshipment. *European Journal of Operational Research*, 235(3):530–539, 2014. doi: 10.1016/j.ejor.2013.10.038.

M.S. Rasmussen, T. Justesen, A. Dohn, and J. Larsen. The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, 219(3):598–610, 2012. ISSN 03772217. doi: 10.1016/j.ejor.2011.10.048.

Sara Reed, Ann Melissa Campbell, and Barrett W. Thomas. The value of autonomous vehicles for last-mile deliveries in urban environments. *Management Science*, 68(1): 280–299, January 2022. doi: 10.1287/mnsc.2020.3917.

U Ritzinger, B Hu, H Koller, and M Dragaschnig. Synchronizing trucks and trailers in a multiresource problem with variable neighborhood search. *Transportation Research Record*, 2610(1):28–34, 2017. doi: 10.3141/2610-04.

Iman Roozbeh, John W. Hearne, and Delaram Pahlevani. A solution approach to the orienteering problem with time windows and synchronisation constraints. *Heliyon*, 6(6): e04202, June 2020. doi: 10.1016/j.heliyon.2020.e04202.

Ann-Kathrin Rothenbächer, Michael Drexl, and Stefan Irnich. Branch-and-price-and-cut for the truck-and-trailer routing problem with time windows. *Transportation Science*, 52(5):1174–1190, October 2018. doi: 10.1287/trsc.2017.0765.

L.-M. Rousseau, M Gendreau, and G Pesant. The synchronized dynamic vehicle dispatching problem. *INFOR*, 51(2):76–83, 2013. doi: 10.3138/infor.51.2.76.

Robert Russell, Wen-Chyuan Chiang, and David Zepeda. Integrating multi-product production and distribution in newspaper logistics. *Computers & Operations Research*, 35 (5):1576–1588, 2008. ISSN 0305-0548. doi: 10.1016/J.COR.2006.09.002.

M. Angélica Salazar-Aguilar, André Langevin, and Gilbert Laporte. Synchronized arc routing for snow plowing operations. *Computers & Operations Research*, 39(7):1432–1440, 2012. ISSN 03050548. doi: 10.1016/j.cor.2011.08.014.

M. Angélica Salazar-Aguilar, André Langevin, and Gilbert Laporte. The synchronized arc and node routing problem: Application to road marking. *Computers & Operations Research*, 40(7):1708–1715, 2013. ISSN 03050548. doi: 10.1016/j.cor.2013.01.007.

Martin W. P. Savelsbergh. The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, 4(2):146–154, 1992. doi: 10.1287/ijoc.4.2.146.

Daniel Schermer, Mahdi Moeini, and Oliver Wendt. A branch-and-cut approach and alternative formulations for the traveling salesman problem with drone. *Networks*, 76(2): 164–186, June 2020. doi: 10.1002/net.21958.

M Schneider and M Drexl. A survey of the standard location-routing problem. *Annals of Operations Research*, 259(1-2):389–414, 2017. doi: 10.1007/s10479-017-2509-0.

Stefan Seuring, Martin Müller, Magnus Westhaus, and Romy Morana. Conducting a literature review — the example of sustainability in supply chains. In *Research Methodologies in Supply Chain Management*, pages 91–106. Physica-Verlag, 2005. doi: 10.1007/3-7908-1636-1_7.

Yong Shi, Yanjie Zhou, Wenhui Ye, and Qian Qian Zhao. A relative robust optimization for a vehicle routing problem with time-window and synchronized visits considering greenhouse gas emissions. *Journal of Cleaner Production*, 275:124112, December 2020. doi: 10.1016/j.jclepro.2020.124112.

Michele D. Simoni, Erhan Kutanoglu, and Christian G. Claudel. Optimization and analysis of a robot-assisted last mile delivery system. *Transportation Research Part E: Logistics and Transportation Review*, 142:102049, October 2020. doi: 10.1016/j.tre.2020.102049.

Hannah Snyder. Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, 104:333–339, 2019. doi: 10.1016/j.jbusres.2019.07.039.

Ricardo Soares, Alexandra Marques, Pedro Amorim, and Jussi Rasinmäki. Multiple vehicle synchronisation in a full truck-load pickup and delivery problem: A case-study in the biomass supply chain. *European Journal of Operational Research*, 277(1):174–194, 2019. doi: 10.1016/j.ejor.2019.02.025.

Felix Tamke and Udo Buscher. A branch-and-cut algorithm for the vehicle routing problem with drones. *Transportation Research Part B: Methodological*, 144:174–203, February 2021. doi: 10.1016/j.trb.2020.11.011.

C Tilk, M Drexl, and S Irnich. Nested branch-and-price-and-cut for vehicle routing problems with multiple resource interdependencies. *European Journal of Operational Research*, 2019. doi: 10.1016/j.ejor.2019.01.041.

Christian Tilk, Nicola Bianchessi, Michael Drexl, Stefan Irnich, and Frank Meisel. Branch-and-price-and-cut for the active-passive vehicle-routing problem. *Transportation Science*, 52(2):300–319, 2018. doi: 10.1287/trsc.2016.0730.

Z Xue, C Zhang, W.-H. Lin, L Miao, and P Yang. A tabu search heuristic for the local container drayage problem under a new operation mode. *Transportation Research Part E: Logistics and Transportation Review*, 62:136–150, 2014. doi: 10.1016/j.tre.2013.12.007.

E Yakıcı, R F Dell, T Hartman, and C McLemore. Daily aircraft routing for amphibious ready groups. *Annals of Operations Research*, 264(1-2):477–498, 2018. doi: 10.1007/s10479-017-2740-8.

Peng-Yeng Yin and Ya-Lan Chuang. Adaptive memory artificial bee colony algorithm for green vehicle routing with cross-docking. *Applied Mathematical Modelling*, 40(21-22):9302–9315, 2016. ISSN 0307-904X. doi: 10.1016/J.APM.2016.06.013.

Vincent F. Yu, Parida Jewpanya, and A.A.N. Perwira Redi. Open vehicle routing problem with cross-docking. *Computers & Industrial Engineering*, 94:6–17, 2016. doi: 10.1016/j.cie.2016.01.018.

Appendix 2.A Classification Table

Table 2.5: Overview of examples of different types of problem aspects in the VRP literature

Problem variant / Application	**Representative Examples**	**Local Aspects**				**Synchronisation Aspects**		**Global Aspects**			**Integration Aspects**
		Cap	***Op***	***TW***	***RL***	***Op***	***Mov***	***Tsk***	***Op***	***Res***	
Capacitated VRP	Laporte et al. [1986]	•						◦			
Pickup and Delivery Problem	Parragh et al. [2008]	•	•					◦			
VRP with Backhauls	Deif and Bodin [1984], Goetschalckx and Jacobs-Blecha [1989]	◦	•					◦			
VRP with synchronised pickup and delivery	Gschwind [2018]	◦	•	◦				◦			
VRP with Time Windows	Savelsbergh [1992]		◦	•				◦			
Distance-constrained VRP	Laporte et al. [1984]			◦	•			◦			
Home Health Care Routing and Scheduling Problem	Bredström and Rönnqvist [2008], Mankowska et al. [2013]			◦		•		◦	•		
Maritime shipping	Bakkehaug et al. [2016], Bélanger et al. [2006]	◦				•		◦	•		
Truck and Trailer Routing Problem	Chao [2002], Parragh and Cordeau [2017]	◦				◦	•	◦	•		
Active-Passive VRP	Meisel and Kopfer [2012], Tilk et al. [2018]		◦	◦		◦	•	◦	•		
VRP with Profits	Chu [2005], Archetti et al. [2009]							•			
Split Delivery VRP	Archetti and Speranza [2012], Irnich et al. [2014]	◦						•		•	
Mixed Fleet VRP	Baldacci et al. [2008]	◦						•		•	
VRP with Cross-Docking	Petersen and Ropke [2011], Yu et al. [2016]	◦		◦				◦		•	
Inventory Routing Problem	Cordeau et al. [2015]			◦				◦			•
Production Routing Problem	Adulyasak et al. [2015], Neves-Moreira et al. [2019]			◦				◦			•

Legend: • - distinctive problem aspect for the problem variant/application; ◦ - problem aspect frequently found but not distinctive for the problem variant/application; Op - Operation constraints; Cap - Capacity constraints; TW - Time Window constraints; RL - Route Length constraints; Mov - Movement constraints; Tsk - Task constraints; Res - Resource constraints.

Appendix 2.B Literature Review Table

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Routing with synchronisation of schedules					
Afifi et al. [2015]	Home Health Care	min TT, max VCP, O	L-TW, **S-O**, G-T, G-O	SA	–
Ait Haddadene et al. [2019]	Home Health Care	min DC, max VCP	**S-O**, G-T, G-O	GA	–
Ali et al. [2021]	Large items delivery and installation	min DC	**S-O**, G-T, L-O	MS, ALNS	–
An et al. [2018]	Construction sector	min FC, OT, TT	**S-O**, G-T, G-O	MS	–
Anaya-Arenas et al. [2019]	Perishable items transportation	min RD	L-O, L-TW, L-RL, **S-O**, G-T, G-O	ILS	–
Anderluh et al. [2017]	Last-mile delivery; Bike delivery	min DC, FC, RD	L-C, L-RL, **S-O**, G-T, G-O	GRASP	–
Anderluh et al. [2019a]	Last-mile delivery; Bike delivery	min DC, FC, RD	L-C, L-RL, **S-O**, G-T, G-O	MCS	U-TT
Anderluh et al. [2019b]	Last-mile delivery; Bike delivery	min DC, FC, RD	L-C, L-RL, **S-O**, G-T, G-O	LNS	–
Bakkehaug et al. [2016]	Ship routing	O	L-O, L-C, L-TW, **S-O**, G-T, G-O	ALNS	–
Bianchessi et al. [2019]	Not specified	min DC, FC, RD	L-C, L-TW, **S-O**, G-T, G-O, L-O	BC	–
Bredström and Rönnqvist [2008]	Home Health Care	min TT	L-TW, **S-O**, G-T, G-O	MH	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Cappanera et al. [2020]	Home Health Care	min DC	L-O, L-TW, L-RL, **S-O**, **S-M**, G-T, G-O	MH	–
Decerle et al. [2018]	Home Health Care	min DC	L-TW, **S-O**, G-T, G-O	MA	–
Dohn et al. [2009]	Not specified	max PT	L-TW, **S-O**, G-T, G-O	DW	–
Dohn et al. [2011]	Not specified	min DC	L-C, L-TW, **S-O**, G-T, G-O	BC, BP	–
Doulabi et al. [2020]	Health Care	min DC, FC, D, WT	L-O, L-C, L-TW, L-RL, **S-O**, L-O	MS, BC	U-TT, U-ST
Emadikhiav et al. [2020]	Calibration of measuring instruments	min TT	L-O, **S-O**, G-T, G-O	BP	–
Fedtke and Boysen [2017]	Railway transportation	min M	L-O, **S-O**, G-T, G-O	DA, H, MS	–
Frifita and Masmoudi [2020]	Home Health Care	min TT	L-TW, **S-O**, G-T, G-O, L-O	VNS	–
Froger et al. [2022]	Charging of electric vehicles	min RD	L-C, L-O, **S-O**, G-T, G-O	MH	–
Ghilas et al. [2016a]	Freight transportation	min DC	L-O, L-C, L-TW, **S-O**, G-T, G-O	ALNS	–
Ghilas et al. [2016b]	Freight transportation	min DC, O	L-O, L-C, L-TW, **S-O**, G-T, G-O	ALNS	–
Goel and Meisel [2013]	Electricity power line maintenance	min DC, O	L-O, **S-O**, G-T, G-O	MH, LNS	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Grimault et al. [2017]	Construction sector	min DC, FC, RD	L-O, **S-O**, G-T, G-O, G-R	ALNS	–
Hà et al. [2020]	Public Utilities; Installation and maintenance services	min DC	L-C, L-TW, **S-O**, G-T, G-O	MS, MH, ALNS	–
Hachemi et al. [2013]	Log truck scheduling	min DC	L-O, **S-O**, **S-M**, G-T, G-O	MS, MH	–
Haddadene et al. [2016]	Home Health Care	max VCP	L-TW, **S-O**, G-T, G-O	GRASP, ILS	–
Hanafi et al. [2019]	Large items delivery and installation	O	L-RL, **S-O**, G-T, G-O	BC	–
Hojabri et al. [2018]	Large items delivery and installation	min DC	L-O, L-TW, **S-O**, G-T, G-O	ALNS	–
Lam and Hentenryck [2016]	Military	min DC	L-C, L-O, L-TW, **S-O**, G-T	BP	–
Li et al. [2020b]	Last-mile delivery	min DC	L-C, L-TW, L-RL, L-O, **S-O**	BC, BP, DW	–
Liu et al. [2019]	Not specified	min DC, FC	L-TW, **S-O**, G-T, G-O	ALNS	–
López-Aguilar et al. [2018]	Not specified	min TT	L-TW, **S-O**, G-T, G-O	MS	–
Luo et al. [2016]	Not specified	min DC, FC	**S-O**, G-T, G-O	BP, BC	–
Mankowska et al. [2013]	Home Health Care	min DC, D, O	L-TW, **S-O**, G-T, G-O	VNS	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Masmoudi et al. [2018]	Home Health Care	O	L-TW, **S-O**, G-T, G-O, L-O	ABC	–
Nguyen et al. [2013]	Not specified	min DC, FC	L-O, L-TW, **S-O**, G-T, G-O	TS	–
Parragh and Doerner [2018]	Home Health Care	min DC, O	**S-O**, G-T, G-O	ALNS	–
Polnik et al. [2020]	Home Health Care	min TT	L-TW, **S-O**, G-T, G-O	MH	–
Pour et al. [2019]	Railway maintenance	min TT	L-TW, **S-O**, G-T, G-O	DA, MS	–
Rasmussen et al. [2012]	Home Health Care	min DC	L-TW, **S-O**, G-T, G-O	DW	–
Roozbeh et al. [2020]	Not specified	O	L-TW, **S-O**, G-T, G-O	ALNS	–
Rousseau et al. [2013]	Not specified	min DC	L-C, L-TW, **S-O**, G-T, G-O	H	D-T
Shi et al. [2020]	Not specified	min DC, FC, O	L-TW, **S-O**, G-T, G-O	SA, TS	U-TT, U-ST
Tilk et al. [2019]	Not specified	min DC, FC, O	**S-O**, G-T, G-O, L-O	BP, BC	–
Routing with transfers or cross-docking requirements					
Brandstätter and Reimann [2018]	Last-mile delivery	min DC, FC, RD	L-O, L-TW, **S-O**, G-T, G-O	H	–
Brandstätter [2019]	Last-mile delivery	min DC, RD	L-C, L-RL, **S-O**, G-T, G-O	ACO, MH	–
Dellaert et al. [2021]	City logistics	min DC, FC	L-TW, **S-O**, G-T, G-O, L-O	BP	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Grangier et al. [2016]	City logistics	min DC, FC	L-O, **S-O**, G-T, G-O	ALNS	–
Grangier et al. [2017]	Cross-docking	min DC	L-O, **S-O**, G-T, G-O	LNS	–
Grangier et al. [2019]	Cross-docking	O	L-O, **S-O**, G-T, G-O, G-R	MH	–
Huang et al. [2018]	Not specified	min DC, FC	L-O, L-C, L-RL, **S-O**, G-T, G-O	ACO	–
Li et al. [2021]	Last-mile delivery	min DC, FC, WT	L-C, L-TW, **S-O**, G-T, L-O	MS, ALNS	–
Marques et al. [2020]	Not specified	min DC, FC	L-C, **S-O**, G-T, G-O, G-R	BC, BP	–
Masson et al. [2013b]	People transportation	min DC	L-C, **S-O**, G-T, G-O	ALNS	–
Masson et al. [2014]	People transportation	min DC	L-O, L-C, L-RL, **S-O**, G-T, G-O	ALNS	–
Medbøen et al. [2020]	Ship routing	min D, O	L-RL, **S-O**, G-T, G-O	MS	U-TT
Medina et al. [2019]	Long-haul and local transportation	min FC, O	L-O, L-C, L-TW, **S-O**, G-T, G-O	MH	–
Mirhedayatian et al. [2019]	Courier transportation	min DC, FC, O	L-O, **S-O**, G-T, G-O, L-O	DA	–
Nolz et al. [2020]	City logistics	min DC	L-C, L-TW, **S-O**, G-T, G-O, L-O	MS	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Peng et al. [2019]	Not specified	min DC, O	L-O, L-C, **S-O**, G-T, G-O	PSO	–
Qu and Bard [2012]	Air transportation	min FC	L-C, L-TW, **S-O**, G-T	GRASP, ALNS	–
Rais et al. [2014]	Not specified	min DC	L-O, L-C, L-TW, **S-O**, G-T, G-O	MS	–
Salazar-Aguilar et al. [2013]	Road marking	min M	L-C, **S-O**, G-T, G-O	ALNS	–
Yin and Chuang [2016]	Not specified	min DC, FC	L-O, L-C, **S-O**, G-T, G-O	ABC	–
Routing with autonomous vehicles					
Agatz et al. [2018]	Last-mile delivery	min DC	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	DP	–
Boysen et al. [2018]	Last-mile delivery; Autonomous robots	min D	L-O, **S-O**, **S-M**, G-T, G-O	MS, ILS	–
Coindreau et al. [2021]	Unmanned aerial vehicles	min DC, TT, M	L-TW, L-RL, **S-O**, **S-M**, G-T, L-O	ALNS	–
Dayarian et al. [2020]	Unmanned aerial vehicles	max PT	L-O, L-TW, **S-O**	MH	D-T
Dell'Amico et al. [2019]	Last-mile delivery; Unmanned aerial vehicles	min RD	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	MS	–
Es Yurek and Ozmutlu [2018]	Last-mile delivery; Unmanned aerial vehicles	min M	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	MH	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Kitjacharoenchai et al. [2019]	Last-mile delivery; Unmanned aerial vehicles	min RD	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	GA	–
Kitjacharoenchai and Lee [2019]	Last-mile delivery; Unmanned aerial vehicles	min TT	L-O, L-C, **S-M**, G-T	MS, LNS	–
Kitjacharoenchai et al. [2020]	Last-mile delivery; Unmanned aerial vehicles	min TT	L-O, L-C, **S-M**, G-T	MS, ALNS	–
Li et al. [2020a]	Last-mile delivery; Unmanned aerial vehicles	min DC, OT, WT	L-C, L-RL, **S-O**, **S-M**, G-T, L-O	MS, ALNS	–
Mourad et al. [2021]	People transportation; Autonomous robots	O	L-C, L-O, **S-O**, G-T, G-O	ALNS	–
Murray and Chu [2015]	Last-mile delivery; Unmanned aerial vehicles	min M	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	H	–
Reed et al. [2022]	Last-mile delivery	min RD	L-O, **S-O**, **S-M**, G-T, G-O	MS	–
Schermer et al. [2020]	Last-mile delivery; Unmanned aerial vehicles	min M	**S-O**, **S-M**, G-T, G-O	BC	–
Simoni et al. [2020]	City logistics.; Autonomous robots	min TT	L-C, **S-O**, G-T, G-O	DP	–
Tamke and Buscher [2021]	Last-mile delivery; Unmanned aerial vehicles	min M	L-O, L-RL, **S-O**, **S-M**, G-T, G-O	BC	–
Routing with trailers or passive vehicles					

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Chao [2002]	Not specified	min DC	L-C, L-O, **S-O**, **S-M**, G-T, G-O, G-R	TS	–
Coindreau et al. [2019]	Energy provider company	min DC	L-RL, **S-O**, **S-M**, G-T, G-O	VNS	–
Derigs et al. [2013]	Not specified	min DC	L-C, L-O, **S-O**, **S-M**, G-T, G-O, G-R	LS, LNS	–
Domínguez-Martín et al. [2018a]	Airline flight scheduling	min DC	**S-O**, **S-M**, G-T, G-O	BC	–
Domínguez-Martín et al. [2018b]	Airline flight scheduling	O	L-RL, **S-O**, **S-M**, G-T, L-O	BC	–
Drexl [2014]	Not specified	min DC, FC	L-C, **S-O**, **S-M**, G-T, G-O	BC	–
Fikar et al. [2016a]	Home Health Care	min TT, WT	**S-O**, **S-M**, G-T, G-O	H	D-T
Fink et al. [2019]	Airport ground handling	min RD, D	**S-O**, **S-M**, G-T, G-O	BP	–
Furian et al. [2018]	Patient transportation	min D	L-O, **S-O**, **S-M**, G-T	H	–
Hu and Wei [2018]	Big-size cargo transportation	min M	**S-O**, **S-M**, G-T, G-O	ACO	–
Meisel and Kopfer [2012]	Container transportation	min DC, RD, max PT	L-O, L-TW, **S-O**, **S-M**, G-T, G-O	ALNS	–
Parragh and Cordeau [2017]	Infrastructure service providers	min DC	L-C, L-O, **S-O**, **S-M**, G-T, G-O, G-R	BP, ALNS	–
Ritzinger et al. [2017]	Container drayage	O	L-O, **S-O**, **S-M**, G-T, G-O	VNS	–

Continued on next page

Table 2.6: Results of the integrative literature review

Reference	Application	Objective Function	Problem aspects	Solution method	Uncertainty and/or dynamism
Rothenbächer et al. [2018]	Raw milk collection	O	L-O, **S-O**, **S-M**, G-T, G-O, G-R	BC, BP	–
Salazar-Aguilar et al. [2012]	Snow plowing	min M	**S-O**, **S-M**, G-O	ALNS	–
Soares et al. [2019]	Biomass residues production and delivery	min DC, FC, RD	L-O, **S-O**, **S-M**, G-T, G-O	MH	–
Tilk et al. [2018]	Not specified	min DC, RD, max PT	L-O, L-TW, **S-O**, **S-M**, G-T, G-O	BP, BC	–
Xue et al. [2014]	Container drayage	min DC, FC	L-O, L-TW, **S-O**, **S-M**, G-T, G-O	TS	–
Yakıcı et al. [2018]	Military	min DC, O	L-RL, **S-O**, **S-M**, G-T, G-O	ACO	–

Legend: Objective Function – DC: Distance costs, FC: Vehicle fixed costs; RD: Route duration; M: Makespan; D: Delays; VCP: Vehicle-customer preferences; TT: Travel time; OT: Operation time; PT: No. of performed tasks; WT: Wait time; O: Other. **Problem aspects** – *Local aspects (L-):* C: Capacity constraints; TW: Time Window constraints; RL: Route Length constraints; *Synchronisation aspects (S-):* O: Operation constraints; M: Movement constraints; *Global aspects (G-):* T: Task constraints; O: Operation constraints; R: Resource constraints. **Solution method** – ABC: Artificial Bee Colony; ACO: Ant Colony Optimisation; ALNS: Adaptive Large Neighbourhood Search; BC: Branch-and-Cut; BP: Branch-and-Price; DA: Decomposition Approach; DP: Dynamic Programming; DW: Dantzig-Wolfe Decomposition; GA: Genetic Algorithm; GRASP: Greedy Randomised Adaptive Search Procedure; H: Heuristic approach; ILS: Iterated Local Search; LNS: Large Neighbourhood Search; MA: Memetic Algorithm; MCS: Monte-Carlo Simulation; MH: Matheuristic; MS: Mathematical Solver (MIP/CP); PSO: Particle Swarm Optimisation; SA: Simulated Annealing; TS: Tabu Search; VNS: Variable Neighbourhood Search; N/A: Not applicable/not provided. **Sources of uncertainty (U)** – TT: Travel times; ST: Service times. **Sources of dynamism (D)** – T: Tasks.

Chapter 3

Modelling and solving routing problems with synchronisation

Multiple vehicle synchronisation in a full truck-load pickup and delivery problem: A case-study in the biomass supply chain

Ricardo Soares *† · **Alexandra Marques*** · **Pedro Amorim***† · **Jussi Rasinmäki**‡

Published in *European Journal of Operational Research*, 2019. DOI: 10.1016/j.ejor.2019.02.025

Abstract The search for higher efficiency in transportation planning processes in real life applications is challenging. The synchronisation of different vehicles performing interrelated operations can enforce a better use of vehicle fleets and decrease travelled distances and non-productive times, leading to a reduction of logistics costs. In this work, the full truck-load pickup and delivery problem with multiple vehicle synchronisation (FT-PDP-mVS) is presented. This problem is motivated by a real-life application in the biomass supply chain "hot-system", where it is necessary to simultaneously perform chipping and transportation operations at the forest roadside. The FT-PDP-mVS consists in determining the integrated routes for three distinct types of vehicles, which need to perform interrelated operations with minimum logistics costs. We extend existing studies in synchronisation of multiple routes by acknowledging several synchronisation aspects, such as operations and movement synchronisation. A novel mixed integer programming model (MIP) is presented, along with valid inequalities to tighten the formulation. A solution method approach is developed based on the fix-and-optimise principles under a variable neighbourhood decomposition search. Results of its application to 19 instances based on a real-world case-study demonstrate its performance. For a baseline instance, the synchronisation aspects tackled in this problem allowed for significant gains when compared to the company's current planning approach. Furthermore, the proposed approach can enhance planning and decision making processes by providing valuable insights about the impact of key parameters of biomass logistics over the routing results.

Keywords transportation · routing · pickup and delivery · synchronisation · OR in natural resources

---

*INESC TEC - Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Porto, Portugal
†Faculdade de Engenharia, Universidade do Porto, Porto, Portugal
‡Simosol Oy, Riihimäki, Finland

3.1. Introduction

Increasing efficiency of transportation planning processes in real life applications is a challenging task. In recent years, there have been many examples of new generation transportation management systems that propose cost-efficient vehicle routes complying with specific business requirements, and without compromising quality of service and reliability. Synchronisation of operations and vehicles' routes can play a major role to increase cost efficiency. Moreover, when the vehicle service time is affected by the existence of other vehicles with similar schedules in the same location, there may be delays or unnecessary idle times that impact the plan flexibility and lead to an increase of the transportation costs.

Different situations can occur in respect to transportation planning under synchronisation constraints. There is often a need for two or more vehicles (or crews) to meet at the same location and time to perform an interrelated operation in order to avoid unnecessary travel efforts, such as additional travel costs, idle times or vehicles. Contrariwise, when there are several vehicles performing similar transportation services, synchronisation at origin and/or destination should prevent queuing and congestion, in order to minimize the idle time. Furthermore, some applications also need to acknowledge the existence of active and passive types of vehicles, where passive vehicles need to be transported by an active vehicle in order to change locations. This paper focuses on performing interrelated operations between vehicles at a given location, as well as ensuring the transportation of passive vehicles through the use of active means of transport. The goal is to find minimum cost routes for distinct types of vehicles engaged in interrelated operations, therefore synchronisation is desirable for minimising the total transportation costs.

This problem is motivated by a real-life application in the biomass supply chain "hot-system", but it has similarities to problems in other sectors. The "hot-system" consists of performing simultaneously chipping and transportation operations at the forest roadside (or pickup location p), as illustrated in Figure 3.1. For this purpose, the vehicle fleet is composed by several trucks (Figure 3.2a), chipping machines (loaders in a general case, Figure 3.2b) and lorries (Figure 3.2c). The chipping machine fragments forest residues into wood chips and directly feeds them into a chargeable container of a truck. Once the container is full, the truck carries the wood chips from the pickup location up to a delivery location d for consumption or storage. Once the pile of forest residues is fully chipped, the chipper is transported to another pickup location p'. Because chippers are often trailer-mounted machines incapable of autonomous movement, they require the service of a lorry. In the end of the planning horizon (usually one day) all vehicles must return to their depot 0. In this framework, synchronisation of the vehicles can reduce chippers and trucks idle times and the costs related with the use of the lorry, therefore decreasing the logistics costs. Since logistics can represent up to 50% of the total cost of the biomass business [Allen et al., 1998], the optimisation of biomass supply operations can in fact improve the cost-efficiency of biomass value chains, ultimately making woody biomass more competitive than alternative fossil fuels.

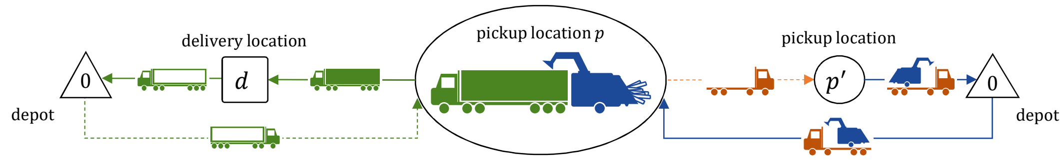


Figure 3.1: Chipping, transportation and hauling network

(a) truck (b) loader (c) lorry

Figure 3.2: Types of vehicles

This real-world application includes a combination of conditions that are not found in the literature. The cost structure of such logistic operations comprises a fixed cost for each vehicle that is used, regardless of its type, as well as two variable cost components, namely the distance travelled by each vehicle and its expended time since the start of the working schedule.

Firstly, the optimal route and schedule for each truck should account for the synchronisation of operations between the loader and the truck at the pickup location. As stated before, both types of vehicles should be present at the same location and at the same time so that the interrelated operations can occur. The loader serves one truck at a time until its container is full, so other trucks that arrive at that pickup location need to queue until the loader becomes again available. The loader remains idle if no truck is present. The literature about the Log-Truck Scheduling Problem (LTSP) addresses many business specificities of wood trucks scheduling, but do not consider these synchronisation aspects (e.g., Hachemi et al. [2009]): typically the loader location is known beforehand and it is unchangeable during the planning horizon, therefore disregarding the chippers transportation cost. Other studies in Vehicle Routing Problems with Multiple Synchronisation Constraints (VRPMSs) do address operations synchronisation to some extent, namely in the prevention of queueing at locations (e.g., Lehuédé et al. [2015], Grimault et al. [2017]), but need to be extended to practical applications in the biomass supply chain, such as ensuring that a loader has been transported and is present at pickup locations.

Secondly, this study addresses the movements of the loader between pickup locations. The route of the loader starts at the depot and encompasses a sequence of pickup locations where there is material to be processed/loaded. The route ends in the depot at the end of the planning horizon, the latest. The loader remains in the same location until all the material there has been loaded in trucks, and only then can be moved. In the end of the horizon, all the pickup locations were visited once by one loader. Moving the loader is a non-productive time that should be avoided. Furthermore, the loader movement also represents extra transportation costs related with the usage of the dedicated lorry that carries the trailer-mounted machine between pickup locations, as well as the variable travel costs,

proportional to the total distance travelled by the lorry. The lorry route starts at the depot and encompasses a sequence of visits to pickup locations for pickup and drop-off of the loaders. Only one loader is transported at a time. After each delivery, the lorry may either engage a new pickup service, come back to the depot empty, or remain in that location. The latter may be convenient in a case of a later pickup service nearby, thus avoiding unnecessary trips with the empty lorry. The studies related with the truck and trailer routing problem (TTRP) already cover some of these business requirements (e.g. Meisel and Kopfer [2012]), but further extensions are needed, as discussed in the next section.
This research builds on a literature review on VRP variants with similarities to our problem. The first contribution of this paper is a new mathematical formulation for integrated route planning for the three types of vehicles (lorries, loaders and trucks), which models the synchronisation constraints based on the approach of Kim et al. [2010]. The model can be solved to optimality for small-scale instances, and a set of valid inequalities is presented to improve its performance. The second contribution of this paper is a matheuristic approach developed to obtain better quality solutions in a reasonable computational time for larger problem instances. The third contribution is to demonstrate the effectiveness of this approach, exemplified in a case-study in the biomass-for-bioenergy supply chain in Finland. For this purpose, the obtained routes and schedules are compared with the outcomes of the current planning approach. Furthermore, valuable managerial insights are provided for planners of the biomass logistics, thus enhancing decision-making processes.
The remainder of this paper is as follows. In Section 3.2, a literature review of similar problems is presented. The problem description is detailed in Section 3.3. In Section 3.4 the novel mathematical formulation and valid inequalities are presented. The proposed solution method for real-world applicable instances is described in Section 3.5. Section 3.6 is devoted to the computational results and managerial insights supported by a case-study in the biomass-for-bioenergy supply chain in Finland. Finally, this paper concludes by summarising the main achievements and future work.

3.2. Literature review

Vehicle Routing Problems (VRPs), firstly introduced by Dantzig and Ramser [1959], aim to find the minimum cost routes for vehicles subjected to a series of constraints. Over the years, several VRP variants were proposed to deal with more realistic applications. The problem in hand covers several aspects of known VRP variants, including the VRP with multiple synchronisation constraints (VRPMS), the pickup and delivery problem (PDP), the log-truck scheduling problem (LTSP) and the truck and trailer routing problem (TTRP). Table 3.1 summarises the literature review of each of the routing problem variants according to each of its features. The following subsections will frame this problem into each one of the enunciated VRP variants. Finally, a brief overview of rich VRPs that mingle some of these VRPs variants will be performed, focusing on problems that show similarities to the one we are tackling.

Table 3.1: Summary table of the literature review

Framework	Problem denomination	Examples of references	Depot		Fleet: Capacity		Fleet: Type of vehicle		Active / passive vehicles		Pickup/delivery requests					Synchronisation			Solution approaches
			Multi	Single	Homogeneous	Heterogeneous	Single	Multiple	Fixed coupling	Flexible coupling	From/to depot	Between customers	Paired requests	Split deliveries	Full truck-loads	Operation simultaneity	Operation precendence	Simultaneous movement	
	OUR PROBLEM		•		•			•		•		•	•	•	•	•	•	•	Matheuristic
VRPMS	HHCRSP: Home Health Care Routing and Scheduling Problem	Bredström and Rönnqvist (2008)		•	•		•									•	•		Matheuristic
VRPMS		Mankowska et al. (2014)		•				•								•	•		Meta-heuristic
VRPMS	CMVRP: Combined man-power vehicle routing problem	Kim et al. (2010)		•			•			•							•	•	Heuristic
GPDP	PDP: Pickup and Delivery Problem	Gendreau et al. (1999)	•		•		•					•	•						Meta-heuristic
GPDP		Korsvik et al. (2011)	•				•					•	•	•	•				Meta-heuristic
GPDP		Ropke and Pisinger (2006) Petersen and Ropke (2011) Ghilas et al. (2016) Gschwind et al. (2018) Ropke et. al (2007) Veenstra et al. (2017)		•		•	•					•	•						Meta-heuristic Column generation Branch-and-price Branch-and-cut
GPDP	LTSP: Log-truck Scheduling Problem	Gronalt and Hirsch (2007) Palmgren et. al. (2004) Flisberg et al. (2009)	•		•		•					•	•		•				Meta-heuristic Column generation Branch-and-price
TTRP	GTTRP: Generalized Truck and Trailer Routing Problem	Drexl (2007) Drexl (2011)		•		•		•		•								•	Branch-and price Branch-and-cut
TTRP	TTRP: Truck and Trailer Routing Problem	Chao (2002) Scheuerer (2006) Lin et al. (2009) Derigs et al. (2013) Parragh and Cordeau (2017)		•	•			•	•								•	•	Meta-heuristic Branch-and-price Branch-and-cut
VRPMS / GPDP	Full truckload pickup and delivery problem with resource synchronisation	Grimault et al. (2017)	•			•	•					•	•	•	•	•	•		Meta-heuristic
VRPMS / GPDP / TTRP	Long-haul freight transportation problem	Neves-Moreira et al. (2016)	•		•			•		•		•	•		•	•		•	Matheuristic
VRPMS / GPDP / TTRP	Active-passive VRP	Meisel and Kopfer (2014) Tilk et al. (2018)		•	•			•		•		•	•		•			•	Meta-heuristic Branch-and-price Branch-and-cut

3.2.1 Vehicle routing problem with multiple synchronisation constraints

The Vehicle Routing Problem with Multiple Synchronisation Constraints (VRPMS), introduced in Drexl [2012], aims to find the minimum cost routes for several vehicles, which need to be synchronised in some nodes to fulfil common tasks. The major distinctive aspect of these problems when compared to traditional VRPs, concerns to the fact that vehicles' routes are dependent of each other. This interdependence leads to complexity when developing solution methods for these types of problems, as a change in one vehicle's route may render all other routes infeasible. In this work, the author distinguishes five types of synchronisation aspects in VRPs, namely load, task, resource, operations (OS) and movement (MS) synchronisation. While load, task and resource synchronisation are intrinsically present in any standard VRP, OS and MS are often dealt separately and in the context of real-life applications. The problem under study can be viewed as a VRP with OS and MS, not only because different vehicles will need to be synchronised in time when arriving at certain locations to perform certain operations, but also because certain vehicles (namely lorries and loaders) will need to move between locations simultaneously.
One of the first works explicitely addressing OS in vehicle routing was Bredström and Rönnqvist [2008]. This work focus on applications in homecare staff scheduling, although it enumerates several alternative applications, such as planning of security guards and forest management. In this problem, each staff member has certain locations to visit, and occasionally customers must be visited by two staff members simultaneously or within a given precedence (e.g., visiting elderly or disabled people at the same time for lifting purposes or with a fixed time offset to apply medical treatments after a meal). From a modelling point of view, the staff members can be viewed as vehicles, or generically resources needed for performing tasks. Besides introducing a Mixed Integer Programming (MIP) model for the problem using a vehicle flow formulation, a solution method is presented, consisting in a heuristic approach where restricted MIP problems are solved iteratively. Kim et al. [2010] present a VRP with OS with the purpose of obtaining a schedule of vehicles that visit a set of customers, where a certain number of tasks has to be performed in a fixed sequence. This problem also intends to assign vehicles to staff, allowing for staff "transshipments" at relay stations in order to respect working time regulations. This paper presents a mathematical formulation based on four-index decision variables x_{ij}^{km}, allowing to account for simultaneous movement of crew m through vehicle k across an arc (i, j). More recently, Mankowska et al. [2014] present a Home Health Care Routing and Scheduling Problem (HHCRSP), which consists in a generalisation of the problem introduced by Bredström and Rönnqvist [2008] where a set of customers needs to be serviced by a set of home carers. However, the specific requirements for the home health care sector are approached differently, as customers may need simultaneous services by more than one home carer at the same time (or with temporal precedence) and not all carers are able to perform certain services. The mathematical model that is presented explicitly handles staff qualifications and simultaneity/precedence requirements. A solution method is also presented, where several neighbourhood structures are devised and used under an Adaptive Variable Neighborhood Search (AVNS) framework.

3.2.2 General pickup and delivery problem

The General Pickup and Delivery Problem (GPDP) consists in a vehicle routing problem where customers pickup and delivery locations are acknowledged and its purpose is to perform the pickup of load at pickup nodes and deliver it to delivery nodes. Parragh et al. [2008a,b] presented a framework for classifying GPDPs, where these problems are split into two main variants, depending on whether the pickup and delivery requests occur from and/or to the depot, or whether the pickups and deliveries occur between customers. The problem under study frames itself in this latter variant and an overview of its characteristics is now provided.

3.2.2.1 Pickup and delivery problem

The Pickup and Delivery Problem (PDP), as it is named by Parragh et al. [2008a,b], is a particularisation of the GPDP. It is the sub-class of GPDPs in which our problem is best framed because it assumes that merchandise from pickup locations must be delivered to a predetermined delivery location.
The PDP is especially relevant in applications where commodities are differentiated and only can be delivered to a specific customer, such as courier transportation [Gendreau et al., 1999] or maritime shipping [Korsvik et al., 2011].
Typical solution approaches for PDPs consist in heuristic methods. One of the most renowned solution approaches is the Adaptive Large Neighborhood Search (ALNS), introduced by Ropke and Pisinger [2006]. The main advantage of the ALNS consists in its flexibility, providing a good balance between diversification and intensification in the search process [Pisinger and Ropke, 2010]. The ALNS is employed for several PDPs [Ropke and Pisinger, 2006, Petersen and Ropke, 2011, Ghilas et al., 2016]. Alternative solution approaches for PDPs include Column-Generation (CG) algorithms [Gschwind et al., 2018] and Branch-and-Price (B&P) and Branch-and-Cut (B&C) algorithms [Ropke et al., 2007, Veenstra et al., 2017].

3.2.2.2 Log-truck scheduling problem

GPDPs can also be classified as less-than-truck-load problems and full-truck-load problems [Parragh et al., 2008b]. The former need to consider capacities of vehicles and the quantities of each node, so that vehicles capacity constraints are met, while the latter does not need to consider quantities, as the delivery unit consists in a single truck-load. When considering the PDP with paired pickups and deliveries as a full-truck-load problem, it is possible to infer that the pickup performed by a truck will necessarily be followed by the delivery of the same load to its corresponding customer. This is a valid assumption in several contexts, namely in the forestry sector, as well as in our case. The Log-truck Scheduling Problem (LTSP) is a particular case of the PDP considering full truck-loads, where previously paired pickup and delivery tasks must be sequenced among the vehicles in order to obtain minimum costs [Gronalt and Hirsch, 2007]. Given the high quantities of raw materials involved in log-truck transportation, the quantities available at one location usually exceed a full truck-load, which also turns this problem into a split delivery PDP. In

the context of our problem, the LTSP resembles the routes done by trucks transporting the merchandise.
For solving the LTSP, Palmgren et al. [2004] presented a B&P algorithm, although Tabu Search (TS) strategies are also common in the literature: Gronalt and Hirsch [2007] uses the unified tabu search (UTS) algorithm [Cordeau et al., 2001] to solve the LTSP and modifies it by allowing neighbourhood size oscillations in the search process. Flisberg et al. [2009] uses a two-phase iterative procedure, where a linear programming problem is solved in the first phase to determine the origin and destination nodes of each full truck-load and, in the second phase, the UTS algorithm is applied for determining the vehicles routes.

3.2.3 Truck and trailer routing problem

Firstly introduced in Chao [2002], the Truck and Trailer Routing Problem (TTRP) encompasses a set of trucks (active vehicles) with coupled trailers (passive vehicles) to perform the necessary deliveries to customers. However, trailers cannot visit all locations, as certain customers have site-dependent vehicle restrictions, so they must first visit a transshipment node where trailers are uncoupled from trucks and loads are transferred from trailers to trucks and/or vice-versa. Afterwards the truck will perform sub-tours to trailer-incompatible customers while leaving the trailer uncoupled at the transshipment node. It is possible to state that the TTRP is a particular case of the VRPMS, due to the occurrence of movement synchronisation between two different types of vehicles.
A typical application of the TTRP is the raw milk collection problem with trucks and trailers, where a dairy company collects raw milk from farmers, and trailers need to be collected by trucks when they are full at milk collection locations. Drexl [2007] performs some extensions to the TTRP, one of them allowing the truck-trailer combinations to be changed within the time horizon, also presenting mathematical formulations using different decision variables: turn variables, designating if a node j is visited immediately after i and immediately before k, arc variables, where routing is set by the arcs that are traversed, and the traditional path variables, where routing is given by whether a node j is visited immediately after node i. In Drexl [2011], the Generalized TTRP (GTTRP) is presented, where time windows and a heterogeneous fleet are introduced.
The TTRP has similarities to our problem when we consider the routes performed by the lorry that moves the loaders between pickup locations. The solution methods for the TTRP present in the literature usually consist in meta-heuristic algorithms, such as TS [Chao, 2002, Scheuerer, 2006], Simulated Annealing (SA) [Lin et al., 2009] and Large Neighborhood Search (LNS) [Derigs et al., 2013]. B&P and B&C algorithms are also rather common [Drexl, 2007, 2011, Parragh and Cordeau, 2017].

3.2.4 Rich vehicle routing problems

Several rich VRPs may be found in the literature that relate to the work in this article, namely in the application of real-world problems. We highlight some of them, due to their intrinsic similarities to our problem.
One of the rich VRPs that exhibits several similarities to our problem is the active-passive

VRP, introduced by Meisel and Kopfer [2012], which consists in a PDP where the fulfilment of the pickup and delivery requests depend on the simultaneous movement of both an active and passive vehicle, such as in a TTRP. Several applications can be found for this problem, such as the transportation of containerised goods. Although the underlying similarities to the TTRP are clear, in this problem there is no pre-defined assignment of active vehicles to passive vehicles, meaning that active vehicles may couple/uncouple any passive vehicle as many times as necessary. Meisel and Kopfer [2012] propose an ALNS approach as a solution method, and since then, exact approaches have also been envisaged in Tilk et al. [2018]. The main distinction between this problem and ours consists in the fact that passive vehicles only have synchronisation requirements with active vehicles. In our case the passive vehicle (loader) not only needs to be synchronised with an active vehicle (lorry), but it is also necessary to synchronise its operations with a third vehicle (truck).
Grimault et al. [2017] presents an ALNS algorithm for a PDP applied to construction, in which construction sites need materials to be delivered from quarries or asphalt concrete plants. Different pickup/delivery requests need to be synchronised, as loading operations require a resource that only loads one vehicle at a time. The main distinction between this problem and ours consists in the assumption that the loading resource is available at each location throughout the entire planning horizon, which does not occur in our case. The problem is solved using instances from a case-study and from the LTSP literature instances in Hirsch [2011]. This problem can therefore be framed not only as a PDP, but also as a VRPMS with resource synchronisation.
Neves-Moreira et al. [2016] presents a long-haul freight transportation problem where tractors are allowed to visit transshipment points in the course of their routes in order to perform exchanges of semi-trailers. The problem is solved using a fix-and-optimise matheuristic for real-world instances. This problem can be framed into any of the problem variants addressed before, as synchronisation is present by acknowledging trucks and trailers, as well as the pickup and delivery nature of the routing problem.

3.3. Problem statement

According to the findings of the literature review, our problem can be considered a full truck-load pickup and delivery problem with multiple vehicle synchronisation (FT-PDP-mVS). In this problem, distinct types of vehicles – trucks and lorries and trailers (henceforth called loaders) – are considered over a transportation network of pickup (supply) and delivery (demand) locations, considering that loaders and lorries are subject to movement synchronisation (MS), and loaders and trucks are subject to operations synchronisation (OS). The considered planning horizon consists in one work day, and all the previously established pickup/delivery requests must be performed in this time frame. The problem objective is to minimize the overall transportation costs, which consist in a trade-off between the number of vehicles used (fixed costs), its travelled distance (distance costs) and the total expended time each vehicle needed to perform its operations (time costs).

3.3.1 Problem entities and features

3.3.1.1 Requests

A request is characterised as a truck-load that needs to be transported from a pickup location to a delivery location. Each truck-load will be delivered to a single delivery location. It is assumed that the commodity assignments from pickup to delivery locations are previously set at a higher planning level, where different objectives from the ones in this paper are considered. Therefore, these assignments are not a decision to be obtained in this problem. Examples of tactical planning problems in the biomass-for-bioenergy supply chain can be found in Marques et al. [2018], Boukherroub et al. [2017], Meyer et al. [2015].
All requests must be performed within the established planning horizon. When the amount required by a delivery location exceeds a full truck-load, requests are split into two or more requests, considering the minimum number of truck-loads needed to satisfy the demand.

3.3.1.2 Vehicles – loaders, lorries and trucks

Loaders are entities that perform an operation on raw materials at pickup locations, prior to its transportation to delivery locations. They are assumed to be homogeneous in terms of its productivity, i.e. the amount of material produced per unit of time. Loaders consist in trailer-mounted machines that are pulled by a lorry whenever they need to be moved (to and from the depot, and between pickup locations).
Lorries are support vehicles and only carry out operations related with the loaders pickup and drop-off. A lorry can only transport one loader at a time but it is able to transport multiple loaders throughout the course of their routes, i.e. different loaders can be coupled to (or decoupled from) lorries as many times as necessary. There may exist some particular cases where this situation does not apply (i.e. lorries and loaders are unique vehicles and cannot be separated, precluding the possibility of a lorry performing loader drop-offs and/or pickups during its route).
Trucks are the vehicles responsible for delivering the truck-loads from the pickup locations to the delivery locations. The trucks fleet is considered homogeneous, i.e. all vehicles exhibit equal transportation capacity.

3.3.1.3 Transportation network

The transportation network encompasses a set of depots, a set of pickup locations and a set of delivery locations.
Depots are starting and ending locations for vehicle routes. All vehicles are associated with a starting depot location, which is previously known, and it is not mandatory that they end their route at that same depot. A depot also serves as a location for picking up and/or dropping off a loader in the course of a lorry route.
Pickup locations are characterised for being geographically dispersed, each having a known amount of material that is available to be transported to delivery locations. The amount of material present at a pickup location typically exceeds the unit capacity of a truck, thus yielding multiple pickup requests per location. Furthermore, the location requires a loader

on site so that trucks' containers can be loaded. As for lorries, they will visit the same pickup location at most twice, one for machine drop-off and another for machine pickup. Due to space availability restrictions at pickup locations, only one loader can be present at the location.

Delivery locations receive cargo that is transported directly from pickup locations by trucks. Trucks may visit the same delivery location more than once in order to deliver multiple truck-loads of material.

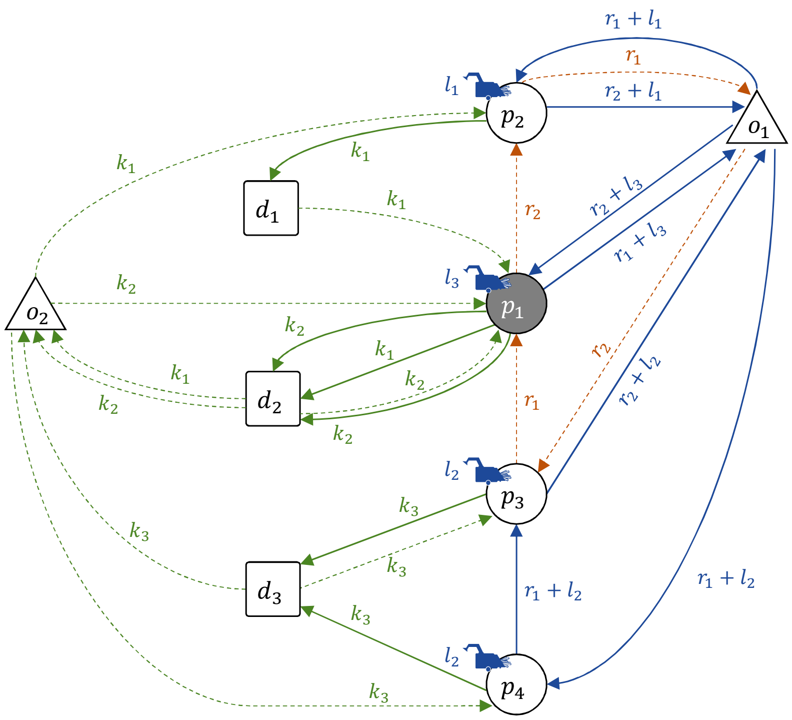
3.3.2 Operations and movement synchronisation

Synchronisation is essential in pickup locations, since loaders need to be transported to the location by lorries (MS), as well as coordinate its temporal availability with both lorries and trucks (OS) so that operations can be performed with the smallest idle time possible. Loaders only visit pickup locations once, and therefore can only leave after all material is loaded into trucks.

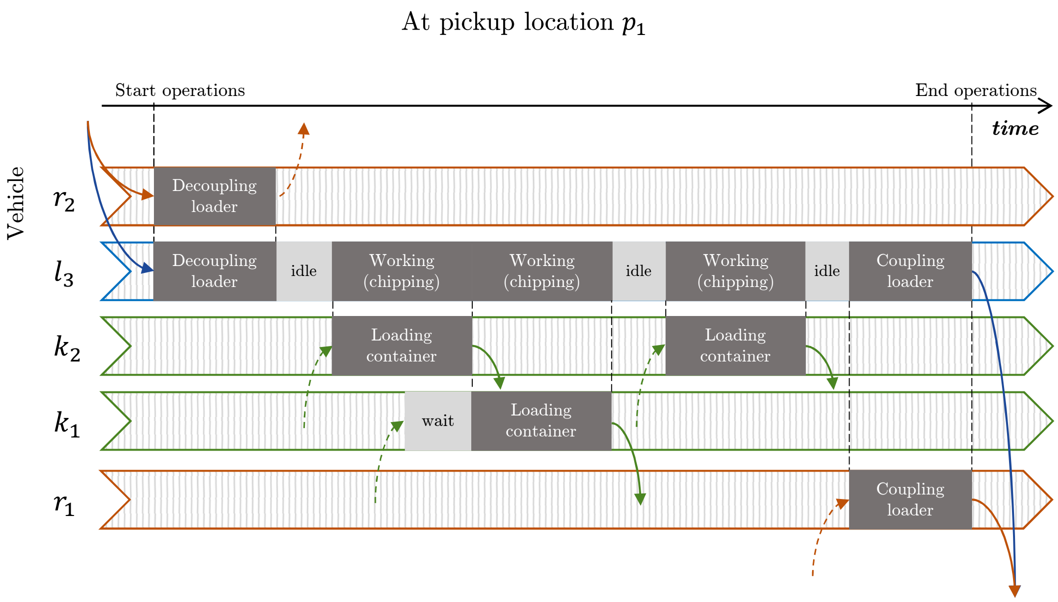
Figure 3.3a presents an example of feasible routes for two lorries, $R = \{r_1, r_2\}$, three loaders, $L = \{l_1, l_2, l_3\}$, and three trucks, $K = \{k_1, k_2, k_3\}$, over a transportation network composed by two depots, $O = \{o_1, o_2\}$, four pickup locations, $P = \{p_1, p_2, p_3, p_4\}$, and three delivery locations, $D = \{d_1, d_2, d_3\}$. An example of a feasible route for truck k_1 is $\{o_2, p_2, d_1, p_1, d_2, o_2\}$. A feasible route for loader l_1 is $\{o_1, p_2, o_1\}$. The lorry r_1 starts the route carrying loader l_1 from the depot to p_2. Then, it returns empty to depot o_1, where it engages with loader l_2 and transports it to p_4. Lorry r_1 waits there until all operations of l_2 are over, and then transports it to p_3. After dropping off loader l_2 in p_3, the lorry will finish its service by collecting loader l_3 at p_1 (left there previously by lorry r_2) and returning it to the depot. In sum, a feasible route of lorry r_1 is $\{o_1, p_2, o_1, p_4, p_3, p_1, o_1\}$. Regarding lorry r_2, a feasible route is $\{o_1, p_1, p_2, o_1, p_3, o_1\}$.

Movement synchronisation is shown in the similarities between the routes of the loader l_1 and lorries r_1 and r_2. The first part of the route of the loader l_1 is coincident with the beginning of the route of lorry r_1, while the second part of the route is coincident with the ending of the route of lorry r_2. The same logic applies to loaders l_2 and l_3.

Operations synchronisation at pickup location p_1 along the time is illustrated in Figure 3.3b. Operations start when lorry r_2 arrives at the location and decouples loader l_3. The first loading operation of loader l_3 starts when truck k_2 arrives. Since truck k_1 arrives while k_2 is still being served, k_1 waits on site and is served right after. After serving truck k_1 there is no other truck available, so loader l_1 remains idle until truck k_2 returns to location p_1. Next, loader l_3 will be moved to another pickup location because the requests (i.e. amount of material available) are equivalent to three full truck-loads, now exhausted. Operations at p_1 end when lorry r_1 engages loader l_3 and departures from p_1. It is noteworthy that operations synchronisation with precedences can also be verified in this example. In fact, decoupling the loader from the lorry must be performed before any loading task with trucks. The same applies to the coupling operation, as it must be the final operation to be performed at the pickup location.



(a) Feasible routes for the three types of vehicles, where $O = \{o_1, o_2\}$ are depots, $P = \{p_1, p_2, p_3, p_4\}$ are pickup locations, $D = \{d_1, d_2, d_3\}$ are delivery locations, $R = \{r_1, r_2\}$ are lorries, $L = \{l_1, l_2, l_3\}$ are loaders and $K = \{k_1, k_2, k_3\}$ are trucks; dotted lines are trips with an empty truck (in green) or lorries without the trailer-mounted loader (in orange)



(b) Operations synchronisation at pickup location p_1

Figure 3.3: Example of routing plan considering multiple vehicle synchronisation

3.4. Problem modelling

The modelling approach encompasses three routing sub-problems, one for each type of vehicle – lorries, loaders and trucks –, all of them intertwined with synchronisation constraints in order to ensure the problem requirements. For each one of the routing sub-problems, a vehicle flow formulation is used, where the generic decision variables state if a given vehicle traverses arc (i, j) or not.

Given that multiple operations may be performed at each location, it is necessary to consider artificial nodes in the transportation network. For the sake of clarity, we use the term “location” to designate a real-world geographical site and the term “node” is used for referring to the nodes of the transportation network, used for modelling purposes. Therefore, a location may have one or more nodes associated with it.

3.4.1 Transportation network preprocessing

A data preprocessing phase is crucial for reducing the model size and complexity, and consequently reducing the solution time when using a commercial solver. This phase consists in the creation of the nodes and arcs in a way that some constraints related with the solution consistency are dealt *a priori* and do not need to be included in the mathematical model.

The network preprocessing phase consists in the creation of artificial nodes and the generation of arcs, explained in the following subsections. Each vehicle type will have its own transportation network, generated according to the specific tasks it is able to perform. To illustrate the rationale behind the network preprocessing, Figure 3.4 revisits the example given in the problem statement section.

3.4.1.1 Depot nodes

Start of vehicles routes: each depot location $o \in O$ contains a start node O_o^+, where vehicle routes will necessarily start, regardless of the type of vehicle. Each vehicle is associated with an initial depot location.

End of vehicles routes: each depot location $o \in O$ contains a sink node O_o^-, where a vehicle ends its route. The sink node that a vehicle visits may not necessarily correspond to the same location it started from. There are no restrictions to the number of vehicles each depot sink node can receive.

Pickups and drop-offs of loaders: each depot location $o \in O$ will have a fixed number n of intermediate nodes $\left\{\hat{O}_o^1, ..., \hat{O}_o^n\right\}$. These nodes are not mandatory to be visited, and their purpose is to allow for a lorry to visit the depot in the middle of its route, either to drop off or to pick up a loader present at that location. The number of intermediate nodes that are created corresponds to the maximum number of pickups and drop-offs the depot will be able to handle.

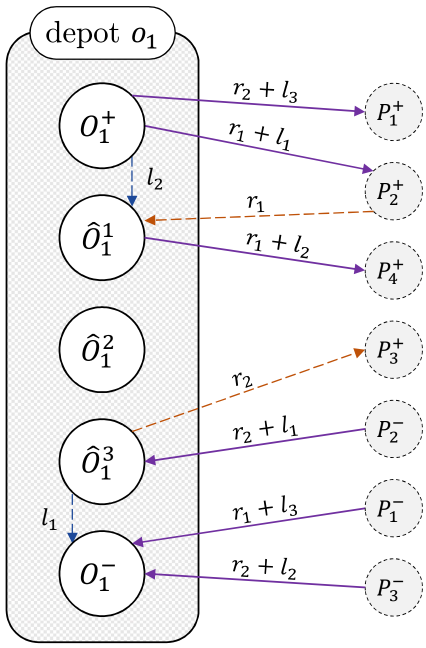
In Figure 3.4a it is possible to verify that lorries $r_1, r_2 \in R$ and loaders $l_1, l_2, l_3 \in L$ all leave node O_1^+, which is the start node of depot o_1. The same applies for trucks $k_1, k_2, k_3 \in K$ in the start node of depot o_2. Also, lorry r_1 visits intermediate node $\hat{O}_1^1$, at where loader l_2 also arrives from O_1^+ and engages with r_1. The inverse situation can also be verified, as lorry r_2 performs the drop-off of loader l_1 at intermediate node $\hat{O}_1^3$, after which loader l_1 finishes its service by visiting the depot sink node and lorry r_2 proceeds with the remainder of its route. All these vehicles end their routes at the sink node of depot o_1.

3.4.1.2 Delivery location nodes

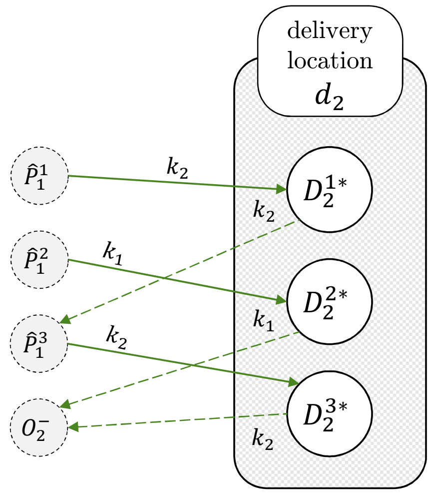
Unloading raw materials: each delivery location $d \in D$ has a fixed number n of unloading nodes, $\left\{D_d^{1\star},...,D_d^{n\star}\right\}$, each one of them corresponding to the reception of a truck-load from its corresponding origin, thus acknowledging the split delivery nature of this problem.
In Figure 3.4b this situation is easily observable, as trucks $k_1, k_2 \in K$ visit delivery location d_2 three times, once for each unloading node $D_2^{1\star}, D_2^{2\star}, D_2^{3\star}$ to deliver the truck-loads originating from pickup location p_1.

3.4.1.3 Pickup location nodes

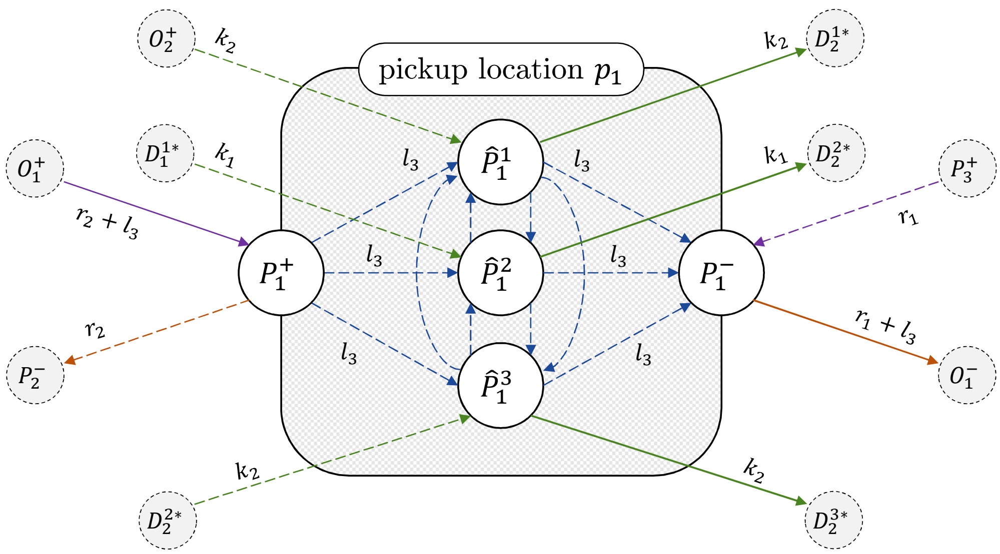
Decoupling loader from lorry: each pickup location $p \in P$ contains a decoupling node P_p^+. These nodes are used for considering movement synchronisation between lorries and loaders when these vehicles arrive at the pickup location simultaneously. After the loader drop-off is complete, the lorry will be able to leave the location and perform other services.
Coupling loader to lorry: each pickup location $p \in P$ contains a coupling node P_p^-. These nodes are used for considering movement synchronisation between lorries and loaders when these vehicles leave the pickup location simultaneously. A lorry will arrive at this node, as well as the loader that was assigned to that location and has finished its service. Arcs to this node are generated taking into account that this node can only be visited when all other nodes of the location have already been visited.
Loading raw materials: each pickup location $p \in P$ has a fixed number n of loading nodes, $\left\{\hat{P}_p^1,...,\hat{P}_p^n\right\}$, which correspond to the number of truck-loads trucks will need to transport from that location (corresponding to the requests defined beforehand). In each of these nodes, a loader and a truck will arrive in order to start loading operations for the truck-load associated with that node.
In Figure 3.4c, pickup location p_1 comprises three loading nodes, corresponding to three truck-loads. Initially lorry r_2 and a loader l_3 arrive simultaneously at decoupling node P_1^+. From this point onwards, the lorry's and loader's routes will diverge, as lorry r_2 will perform the pickup of another loader at pickup location p_2, and loader l_3 will remain at location p_1 until all work is complete. The loader will "travel" to one of the possible loading nodes $\left\{\hat{P}_1^1, \hat{P}_1^2, \hat{P}_1^3\right\}$ to begin loading operations. Trucks will also arrive at the same node where loader l_3 has gone to, ensuring that loading operations can start. Afterwards the truck leaves the location to deliver the merchandise, as loader l_3 continues to travel to the remaining loading nodes. Finally, when all loading nodes are visited (and therefore all work is complete), the loader will visit the coupling node P_1^-, where lorry r_1 arrives to pick it up and finish their routes by visiting the sink node of depot o_1.



(a) For depot location $o_1 \in O$



(b) For delivery location $d_2 \in D$



(c) For pickup location $p_1 \in P$

Figure 3.4: Node and arc generation

3.4.1.4 Arc generation

The arc generation rules, synthesised in Table 3.2, aim to remove arc possibilities from the transportation network that would lead to inconsistencies. For example, arcs where a lorry and a loader leave a decoupling node may be eliminated, since the lorry must have already left the loader in the location.

Table 3.2: Arc generation for lorries, r, loaders, l, and trucks, k

Origin \ Destination		Depots: Start node, O^+	Depots: Intermediate nodes, $\hat{O}$	Depots: Sink nodes, O^-	Pickup locations: Coupling nodes, P^+	Pickup locations: Loading nodes, $\hat{P}$	Pickup locations: Decoupling nodes, P^-	**Delivery locations** Unloading nodes, D^*
Depots	Start node, O^+		$l^\#$		$r+l$	k	r	
	Intermediate nodes, $\hat{O}$			$l^\#$	$r+l$		r	
	Sink nodes, O^-							
Pickup locations	Coupling nodes, P^+		r	r		$l^\#$	r	
	Loading nodes, $\hat{P}$					$l^\#$	$l^\#$	k
	Decoupling nodes, P^-		$r+l$	$r+l$	$r+l$			
Delivery locations Unloading nodes, D^*				k		k		

- within the same location

It is worth noting that arc generation for lorries and loaders present in Table 3.2 is valid for the general case, where loaders may be decoupled from lorries. For cases where a lorry and a loader act as an unique vehicle and cannot be split, movement synchronisation must be met at all times, and decision variables for these vehicles only need to be generated this vehicle combination. In these cases, the network can be significantly simplified. For example, arcs heading and originating from depot intermediate nodes would be removed, and lorries would necessarily go from a decoupling node to the coupling node of the same location, i.e. cannot leave the pickup location as long as the loader does not finish all of its loading tasks.

In the event that all lorries and loaders act as unique vehicles, depot intermediate nodes would have no use in the model, and therefore they would be eliminated.

3.4.2 Mathematical formulation

The mathematical formulation of the mixed integer programming model for the FT-PDP-mVS is now presented. The sets and parameters will firstly be presented, followed by the decision variables, the objective function and the model constraints.

3.4.2.1 Sets and parameters

Most of the model sets have already been presented, but they are presented again for the sake of clarification.
Locations and nodes: the sets containing locations and nodes are the following. In order to simplify notation throughout the formulation, sets of nodes may be added a subscript relating to a specific location, in which case we only refer to nodes of that specific location.

O	Set of depot locations
P	Set of pickup locations
D	Set of delivery locations
$O^+, \hat{O}, O^-$	Set of depot start, intermediate and sink nodes, respectively
$P^+, \hat{P}, P^-$	Set of pickup decoupling, loading and coupling nodes, respectively
$D^\star$	Set of delivery unloading nodes

Auxiliary sets are also defined for depot locations, $O^\star = O^+ \cup \hat{O} \cup O^-$, and pickup locations, $P^\star = P^+ \cup \hat{P} \cup P^-$, for designating the sets of all depot and pickup nodes, respectively.
Vehicles: let R, L and K be the sets of lorries, loaders and trucks, respectively. Also, let σ and χ be a fictitious lorry and loader, respectively, and with that in mind, $R^\star = R \cup \{\sigma\}$ and $L^\star = L \cup \{\chi\}$. These fictitious vehicles are used for modelling arcs where one travels without the other. Sets $R^\star$ and $L^\star$ are used for defining the model's decision variables, which are described in this subsection. For designating all real (i.e. non-fictitious) vehicles, set V is also defined, $V = R \cup L \cup K$.
Directed graphs: the problem as a whole is defined through two directed incomplete graphs. For lorries and loaders, set $G = (N, A)$, with $N = O^\star \cup P^\star$ is used. Let δ^+_{irl} and δ^-_{irl} be the sets of nodes that follow and precede node $i \in N$ when using vehicles $r \in R^\star$ and $l \in L^\star$, respectively. The directed graph for trucks is $G' = (N', A')$, with $N' = O^+ \cup O^- \cup \hat{P} \cup D^\star$, and φ^+_{ik} and φ^-_{ik} define the sets of nodes that follow and precede $i \in N'$ when using vehicle $k \in K$, respectively. As expected, if a given node i is not possible to be visited by a vehicle (or pair of vehicles) due to the network preprocessing, sets $\varphi^+_{ik}(\delta^+_{irl})$ and $\varphi^-_{ik}(\delta^-_{irl})$ will result in an empty set.
Parameters: the problem parameters are defined as follows.

C_v	Fixed usage cost of vehicle $v \in V$
c'_v	Usage cost per distance unit of vehicle $v \in V$
d'_{ij}	Travel distance between node $i \in (N \cup N')$ and node $j \in (N \cup N')$
c_v	Usage cost per time unit of vehicle $v \in V$
d_{ij}	Travel time between node $i \in (N \cup N')$ and node $j \in (N \cup N')$
s_i	Service time of operation being performed at node $i \in (N \cup N')$
T	Duration of the planning horizon

Note that parameter s_i relates to the necessary time for each operation to be performed by a vehicle in a specific node, e.g., the time a loader takes to load a specific truck-load, or the time a truck needs to perform the unloading operation at unloading nodes.

3.4.2.2 Decision variables

For each one of the routing sub-problems, a vehicle flow formulation is used. For the sub-problem concerning the trucks routes, a standard set of binary three-index path decision variables was considered. In respect to lorries and loaders, a new set of decision variables was needed to address movement synchronisation aspects. To that effect, a similar modelling approach to the one in Kim et al. [2010] is adopted. Consider a unique set of binary four-index path decision variables x_{ij}^{rl}, taking the value 1 if arc (i, j) is traversed by lorry r and loader l simultaneously, 0 otherwise. For situations where the lorry moves without the loader, for example, a fictitious entity replaces the index of the missing entity. Therefore, $x_{ij}^{r\chi}$ would mean that lorry r traverses arc (i, j) carrying the fictitious loader χ. By using these decision variables, movement synchronisation is easily introduced (or removed) in the model, thus avoiding the addition of unnecessary synchronisation constraints.
Additionally, due to the need to ensure synchronisation requirements, it is necessary to define additional decision variables which will account for the arrival time of vehicles to locations.
Therefore, the models decision variables are the ones that follow.

$$x_{ij}^{rl} \quad \begin{cases} 1 & \text{if arc } (i, j) \in A \text{ is traversed by lorry } r \in R^{\star} \text{ and loader } l \in L^{\star} \text{ simultaneously;} \\ 0 & \text{otherwise.} \end{cases}$$

$$y_{ij}^{k} \quad \begin{cases} 1 & \text{if arc } (i, j) \in A' \text{ is traversed by truck } k \in K; \\ 0 & \text{otherwise.} \end{cases}$$

t_{iv} Arrival time of vehicle $v \in V$ at location $i \in (N \cup N')$

3.4.2.3 Objective function

The objective function (equation (3.1)) consists in minimising the overall costs. It contains three main components: equation (3.1a) corresponds to the sum of fixed costs for each one of the types of vehicles (lorries, loaders and trucks), and equation (3.1b) corresponds to the costs associated with the travelled distance for each type of vehicle. Equation (3.1c) corresponds to vehicle time costs: it accounts for the total time each vehicle spent in performing its routes and therefore is computed by using the arrival time of each vehicle at depot sink nodes. This summand is the one that guides the MIP model towards the minimisation of the makespan of each vehicle, which corresponds to the elapsed time since the start of the work day up to the time instant it finishes its route. All vehicles start the work day at time 0.

$$\min \sum_{r\in R}\sum_{l\in L^\star}\sum_{j\in O^-}\sum_{i\in\delta^-_{jrl}} C_r\cdot x^{rl}_{ij} \quad +\sum_{l\in L}\sum_{r\in R^\star}\sum_{j\in O^-}\sum_{i\in\delta^-_{jrl}} C_l\cdot x^{rl}_{ij} \quad +\sum_{k\in K}\sum_{j\in O^-}\sum_{i\in\varphi^-_{jk}} C_k\cdot y^k_{ij} \tag{3.1a}$$

$$+\sum_{r\in R}\sum_{l\in L^\star}\sum_{(i,j)\in A} d'_{ij}\cdot c'_r\cdot x^{rl}_{ij} \quad +\sum_{l\in L}\sum_{r\in R^\star}\sum_{(i,j)\in A} d'_{ij}\cdot c'_l\cdot x^{rl}_{ij} \quad +\sum_{k\in K}\sum_{(i,j)\in A'} d'_{ij}\cdot c'_k\cdot y^k_{ij} \tag{3.1b}$$

$$+\sum_{v\in V}\sum_{j\in O^-} c_v\cdot t_{jv} \tag{3.1c}$$

3.4.2.4 Constraints

The constraints directly related to lorries correspond to equations (3.2)-(3.8). Constraints (3.2)-(3.3) guarantee the depot is the start and end of all lorry routes. Constraints (3.4) ensure the intermediate depot nodes are only visited by lorries if they have already left the depot start node. Constraints (3.5) establish the route inflow-outflow conditions for lorries. Constraints (3.6)-(3.7) set the lorries arrival times at nodes. Constraints (3.8) are linking constraints between the arrival times at nodes and the assigned routes to lorries.

$$\sum_{l\in L^\star}\sum_{i\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} = \sum_{l\in L^\star}\sum_{j\in O^-}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} \qquad \forall r\in R \tag{3.2}$$

$$\sum_{l\in L^\star}\sum_{i\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} \le 1 \qquad \forall r\in R \tag{3.3}$$

$$\sum_{l\in L^\star}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} \le \sum_{l\in L^\star}\sum_{o\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{oj} \qquad \forall r\in R, \forall i\in\hat{O} \tag{3.4}$$

$$\sum_{l\in L^\star}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} = \sum_{l\in L^\star}\sum_{i\in\delta^+_{jrl}} x^{rl}_{ji} \qquad \forall r\in R, \forall j\in N\setminus(O^+\cup O^-) \tag{3.5}$$

$$s_i + d_{ij} \le t_{jr} + M\cdot\left(1-\sum_{l\in L^\star} x^{rl}_{ij}\right) \qquad \forall r\in R, \forall i\in O^+, \forall j\in N\setminus O^+ \tag{3.6}$$

$$t_{ir} + s_i + d_{ij} \le t_{jr} + M\cdot\left(1-\sum_{l\in L^\star} x^{rl}_{ij}\right) \qquad \forall r\in R, \forall i\notin O^+, \forall j\in N\setminus O^+ \tag{3.7}$$

$$t_{jr} \le M\cdot\sum_{l\in L^\star}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} \qquad \forall r\in R, \forall j\in N\setminus O^+ \tag{3.8}$$

The constraints related to loaders correspond to equations (3.9)-(3.15) and are now introduced. Constraints (3.9)-(3.10) guarantee the depot is the start and end of all loader routes. Constraints (3.11) ensure the intermediate depot nodes are only visited by loaders if they have already left the depot start node. Constraints (3.12) establish the route inflow-outflow conditions for loaders. Constraints (3.13)-(3.14) set the loaders arrival times at nodes. Constraints (3.15) are linking constraints between the arrival times at nodes and the assigned routes to loaders.

$$\sum_{r\in R^\star}\sum_{i\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} = \sum_{r\in R^\star}\sum_{j\in O^-}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} \qquad \forall l\in L \tag{3.9}$$

$$\sum_{r\in R^\star}\sum_{i\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} \leq 1 \qquad \forall l\in L \tag{3.10}$$

$$\sum_{r\in R^\star}\sum_{j\in\delta^+_{irl}} x^{rl}_{ij} \leq \sum_{r\in R^\star}\sum_{o\in O^+}\sum_{j\in\delta^+_{irl}} x^{rl}_{oj} \qquad \forall l\in L, \forall i\in\hat{O} \tag{3.11}$$

$$\sum_{r\in R^\star}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} = \sum_{r\in R^\star}\sum_{i\in\delta^+_{jrl}} x^{rl}_{ji} \qquad \forall l\in L, \forall j\in N\setminus(O^+\cup O^-) \tag{3.12}$$

$$s_i + d_{ij} \leq t_{jl} + M\cdot\left(1-\sum_{r\in R^\star} x^{rl}_{ij}\right) \qquad \forall l\in L, \forall i\in O^+, \forall j\in N\setminus O^+ \tag{3.13}$$

$$t_{il} + s_i + d_{ij} \leq t_{jl} + M\cdot\left(1-\sum_{r\in R^\star} x^{rl}_{ij}\right) \qquad \forall l\in L, \forall i\notin O^+, \forall j\in N\setminus O^+ \tag{3.14}$$

$$t_{jl} \leq M\cdot\sum_{r\in R^\star}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} \qquad \forall l\in L, \forall j\in N\setminus O^+ \tag{3.15}$$

The constraints related to trucks correspond to equations (3.16)-(3.21) and are the ones that follow. Constraints (3.16)-(3.17) guarantee the depot is the start and end of all truck routes. Constraints (3.18) establish the route inflow-outflow conditions for trucks. Constraints (3.19)-(3.20) set the trucks arrival times at nodes. Constraints (3.21) are linking constraints between the arrival times at nodes and the assigned routes to trucks.

$$\sum_{i\in O^+}\sum_{j\in\varphi^+_{ik}} y^k_{ij} = \sum_{j\in O^-}\sum_{i\in\varphi^-_{jk}} y^k_{ij} \qquad \forall k\in K \tag{3.16}$$

$$\sum_{i\in O^+}\sum_{j\in\varphi^+_{ik}} y^k_{ij} \leq 1 \qquad \forall k\in K \tag{3.17}$$

$$\sum_{i\in\varphi^-_{jk}} y^k_{ij} = \sum_{i\in\varphi^+_{jk}} y^k_{ji} \qquad \forall k\in K, \forall j\in N'\setminus(O^+\cup O^-) \tag{3.18}$$

$$s_i + d_{ij} \leq t_{jk} + M\cdot\left(1-y^k_{ij}\right) \qquad \forall k\in K, \forall i\in O^+, \forall j\in N'\setminus O^+ \tag{3.19}$$

$$t_{ik} + s_i + d_{ij} \leq t_{jk} + M\cdot\left(1-y^k_{ij}\right) \qquad \forall k\in K, \forall i\notin O^+, \forall j\in N'\setminus O^+ \tag{3.20}$$

$$t_{jk} \leq M\cdot\sum_{i\in\varphi^-_{jk}} y^k_{ij} \qquad \forall k\in K, \forall j\in N'\setminus O^+ \tag{3.21}$$

The following equations (3.22)-(3.27) establish that all pickup and delivery nodes are visited by its appropriate vehicles exactly once. Constraints (3.22) ensure that exactly one lorry and one loader arrive at decoupling nodes at pickup locations. Constraints (3.23) and (3.24) guarantee that exactly one loader and one truck can visit loading nodes, respectively. Constraints (3.25) and (3.26) ensure that exactly one lorry and one loader can visit coupling nodes, respectively. Finally, constraints (3.27) guarantee that all loading and unloading nodes are visited exactly once by a truck.

$$\sum_{r\in R}\sum_{l\in L}\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} = 1 \qquad \forall j \in P^+ \tag{3.22}$$

$$\sum_{l\in L}\sum_{i\in\delta^-_{j\sigma l}} x^{\sigma l}_{ij} = 1 \qquad \forall j \in \hat{P} \tag{3.23}$$

$$\sum_{k\in K}\sum_{i\in\varphi^-_{jk}} y^{k}_{ij} = 1 \qquad \forall j \in \hat{P} \tag{3.24}$$

$$\sum_{r\in R}\sum_{i\in\delta^-_{jr\chi}} x^{r\chi}_{ij} = 1 \qquad \forall j \in P^- \tag{3.25}$$

$$\sum_{l\in L}\sum_{i\in\delta^-_{j\sigma l}} x^{\sigma l}_{ij} = 1 \qquad \forall j \in P^- \tag{3.26}$$

$$\sum_{k\in K}\sum_{i\in\varphi^-_{jk}} y^{k}_{ij} = 1 \qquad \forall j \in (P^\star \cup D^\star) \tag{3.27}$$

Constraints (3.28) are sub-tour elimination constraints. They avoid lorries and loaders from performing sub-tours through depot intermediate nodes and travelling a lower distance than from visiting a location directly. Depending on how the distance matrix of the problem is conceived, these constraints may or not be needed. If euclidean distances (both for time and distance) are considered, these constraints are not necessary, as vehicles will always prefer to travel a lower distance. If real distance matrices are considered where the objective is to minimize travel time or distance, these constraints should be introduced into the model, as in these situations the triangle inequality assumption does not hold. With these additional constraints, vehicles will not be able to visit an intermediate depot node without either performing a loader drop-off or a pickup, i.e. vehicles cannot leave these nodes in the same form as they arrived.

$$\sum_{i\in\delta^-_{jrl}} x^{rl}_{ij} + \sum_{i\in\delta^+_{jrl}} x^{rl}_{ji} \le 1 \qquad \forall j \in \hat{O}, \forall r \in R^\star, \forall l \in L^\star \tag{3.28}$$

The synchronisation constraints that link the different routing sub-problems correspond to equations (3.29)-(3.30).

$$\sum_{r\in R} t_{ir} = \sum_{l\in L} t_{il} \qquad \forall i \in (P^+ \cup P^-) \tag{3.29}$$

$$\sum_{l \in L} t_{il} = \sum_{k \in K} t_{ik} \qquad \forall i \in \hat{P} \tag{3.30}$$

Constraints (3.29) establish that lorries and loaders must arrive to the pickup decoupling and coupling nodes at the same time and therefore must wait in their previous nodes until both are available to arrive at the same time. Constraints (3.30) are analogous to constraints (3.29) and ensure that trucks arrive at pile loading nodes at the same time as loaders.
Finally, decision variables domain is presented in equations (3.31):

$$x_{ij}^{rl} \in \{0,1\}, \qquad y_{ij}^{k} \in \{0,1\}, \qquad 0 \leq t_{iv} \leq T. \tag{3.31}$$

3.4.2.5 Valid inequalities

We now present valid inequalities that can tighten the mathematical formulation and induce improvements in the branch-and-bound procedure. All of the presented inequalities can be added statically with the model constraints.
One of the three sets of valid inequalities that follow can be used to reduce the occurrence of symmetric solutions on the vehicle allocation. They are VRP-specific valid inequalities adapted from the literature (e.g. Sherali and Smith [2001], Adulyasak et al. [2014]). Assuming that $R_o = \{r_1, r_2, ..., r_{|R_o|}\}$, $L_o = \{l_1, l_2, ..., l_{|L_o|}\}$, $K_o = \{k_1, k_2, ..., k_{|K_o|}\}$ define the sets of lorries, loaders and trucks that depart from depot $o \in O$, respectively, and that the cost structures (i.e. fixed and variable costs) of each vehicle are equal to the ones of its peers, the symmetry-breaking constraints SB1–SB3 that follow can be applied between ordered pairs of vehicles, according to the desired vehicle allocation order.
The first set of symmetry-breaking constraints (SB1) is presented in equations (3.32)-(3.34) and it specifies that a certain vehicle can only be used if another identical vehicle is already being used.

$$\sum_{l \in L^\star} \sum_{i \in O^+} \sum_{j \in \delta^+_{ir_{(n+1)}l}} x_{ij}^{r_{(n+1)}l} \leq \sum_{l \in L^\star} \sum_{i \in O^+} \sum_{j \in \delta^+_{ir_n l}} x_{ij}^{r_n l} \qquad \forall o \in O, 1 \leq n \leq |R_o| - 1 \tag{3.32}$$

$$\sum_{r \in R^\star} \sum_{i \in O^+} \sum_{j \in \delta^+_{irl_{(n+1)}}} x_{ij}^{rl_{(n+1)}} \leq \sum_{r \in R^\star} \sum_{i \in O^+} \sum_{j \in \delta^+_{irl_n}} x_{ij}^{rl_n} \qquad \forall o \in O, 1 \leq n \leq |L_o| - 1 \tag{3.33}$$

$$\sum_{i \in O^+} \sum_{j \in \varphi^+_{ik_{(n+1)}}} y_{ij}^{k_{(n+1)}} \leq \sum_{i \in O^+} \sum_{j \in \varphi^+_{ik_n}} y_{ij}^{k_n} \qquad \forall o \in O, 1 \leq n \leq |K_o| - 1 \tag{3.34}$$

The second set of symmetry-breaking constraints (SB2) is presented in equations (3.35)-(3.37), which consist in alternative tighter constraints that order vehicle allocation by the number of traversed arcs by each vehicle. These constraints further reduce the number of symmetric solutions, compared with SB1.

$$\sum_{l \in L^\star} \sum_{(i,j) \in A} x_{ij}^{r_{(n+1)}l} \leq \sum_{l \in L^\star} \sum_{(i,j) \in A} x_{ij}^{r_n l} \qquad \forall o \in O, 1 \leq n \leq |R_o| - 1 \tag{3.35}$$

$$\sum_{r\in R^\star}\sum_{(i,j)\in A} x_{ij}^{rl_{(n+1)}} \leq \sum_{r\in R^\star}\sum_{(i,j)\in A} x_{ij}^{rl_n} \qquad \forall o \in O, 1 \leq n \leq |L_o|-1 \tag{3.36}$$

$$\sum_{(i,j)\in A'} y_{ij}^{k_{(n+1)}} \leq \sum_{(i,j)\in A'} y_{ij}^{k_n} \qquad \forall o \in O, 1 \leq n \leq |K_o|-1 \tag{3.37}$$

The third set of symmetry-breaking constraints (SB3), presented in equations (3.38)-(3.40), also reduces the number of symmetric solutions by ordering allocated vehicles by their travelled distance in the solution.

$$\sum_{l\in L^\star}\sum_{(i,j)\in A} d'_{ij}\cdot x_{ij}^{r_{(n+1)}l} \leq \sum_{l\in L^\star}\sum_{(i,j)\in A} d'_{ij}\cdot x_{ij}^{r_nl} \qquad \forall o \in O, 1 \leq n \leq |R_o|-1 \tag{3.38}$$

$$\sum_{r\in R^\star}\sum_{(i,j)\in A} d'_{ij}\cdot x_{ij}^{rl_{(n+1)}} \leq \sum_{r\in R^\star}\sum_{(i,j)\in A} d'_{ij}\cdot x_{ij}^{rl_n} \qquad \forall o \in O, 1 \leq n \leq |L_o|-1 \tag{3.39}$$

$$\sum_{(i,j)\in A'} d'_{ij}\cdot y_{ij}^{k_{(n+1)}} \leq \sum_{(i,j)\in A'} d'_{ij}\cdot y_{ij}^{k_n} \qquad \forall o \in O, 1 \leq n \leq |K_o|-1 \tag{3.40}$$

Besides breaking solution symmetry by ordering vehicle allocation, it is also possible to order identical pickup and delivery requests. In fact, if the total quantity demanded by a delivery location coming from a certain pickup location exceeds a full truck-load, the arrival times of vehicles of loading and unloading nodes can be established in a given order. Note that this only applies to truck-load requests whose associated quantities are equal, as well as its origin and destination locations. This fourth set of symmetry-breaking constraints (SB4) is presented in equations (3.41)-(3.42).

$$\sum_{v\in(L\cup K)} t_{iv} \leq \sum_{v\in(L\cup K)} t_{i'v} \qquad \forall p \in P, i, i' \in \hat{P}_p \tag{3.41}$$

$$\sum_{k\in K} t_{jk} \leq \sum_{k\in K} t_{j'k} \qquad \forall d \in D, j, j' \in D_d^\star \tag{3.42}$$

Finally, we present a set of valid inequalities whose purpose is to set lower bounds (LB1) to vehicles arrival times at depot sink nodes. The principles behind these valid inequalities are presented in Meisel and Kopfer [2012] and were adapted to our specific case. These constraints are present in equations (3.43)-(3.45), taking into account the minimum required travel and service times for each vehicle type.

For lorries, constraints (3.43) take into account the minimum required travel times for arriving and leaving a pickup location, plus the service times for performing the decoupling and coupling operations of a loader. These lower bounds become active if lorries visit these pickup locations.

For loaders, constraints (3.44) take into account the minimum required travel times for arriving and leaving a pickup location, plus the service times at loading, decoupling and coupling nodes. These lower bounds become active if loaders visit these pickup locations.

For trucks, constraints (3.45) take into account the minimum required travel times for arriving at a pickup location of a specific request, the service time for the request's loading node, the travel time from the pickup and the delivery location, the service time for the request's unloading node and the minimum required travel time when departing from the delivery location. These lower bounds become active if trucks perform a specific request.

$$\sum_{j \in O^-} t_{jr} \geq \sum_{l \in L} \sum_{p \in P} \sum_{i \in \delta^-_{P^+_p rl}} \min_{i \in N} \left\{ d_{ip} + s_{P^+_p} \right\} \cdot x^{rl}_{iP^+_p} + $$
$$+ \sum_{l \in L} \sum_{p \in P} \sum_{j \in \delta^+_{P^-_p rl}} \min_{i \in N} \left\{ d_{ip} + s_{P^-_p} \right\} \cdot x^{rl}_{P^-_p j} \qquad \forall r \in R \qquad (3.43)$$

$$\sum_{j \in O^-} t_{jl} \geq \sum_{r \in R} \sum_{p \in P} \sum_{i \in \delta^-_{P^+_p rl}} \left(\min_{j,j' \in N} \left\{ d_{jp} + d_{pj'} \right\} + \sum_{p' \in P^\star_p} s_{p'} \right) \cdot x^{rl}_{iP^+_p} \qquad \forall l \in L \qquad (3.44)$$

$$\sum_{j \in O^-} t_{jk} \geq \sum_{p \in \hat{P}} \sum_{d \in D^\star} \min_{i,j \in N} \left\{ d_{ip} + s_p + d_{pd} + s_d + d_{dj} \right\} \cdot y^k_{pd} \qquad \forall k \in K \qquad (3.45)$$

It should be noted that symmetry-breaking constraints SB1–SB3 cannot be combined in the same model, as they would render it infeasible. On the other hand, constraints SB4 and LB1 may be combined with any of the other presented inequalities.

3.5. Matheuristic solution method

Routing problems with inter-vehicle synchronisation have an additional difficulty when developing solution methods, as changing a vehicle's route may have negative effects in the feasibility of other routes.

The solution approach proposed comprises two main phases. The first phase consists in obtaining an initial solution, where slight adaptations to the MIP model need to be considered. The second phase consists in the implementation of a fix-and-optimise matheuristic approach grounded on the principles of a variable neighbourhood decomposition search (VNDS). The following subsections will outline these two phases.

3.5.1 Initial solution

Due to all the intricacies present in the problem at hand and in VRPs with synchronisation in general, the main bottleneck when developing solution methods for large instances of these problems consists in solution feasibility.

The initial tests performed with the largest case-study instances showed that the commercial solver had difficulties obtaining an initial solution. For some cases the solver was not able to find any feasible solution within a two hour time limit. In order to overcome this problem and to allow a fast convergence into the feasible domain, high penalty costs for unaccomplished pickup/delivery requests were introduced into the objective function, thus allowing infeasible solutions to be considered in early stages of the algorithm.

The addition of penalties for unaccomplished requests has some repercussions in the model described in the previous section. Constraints (3.22)-(3.27), which ensure that all nodes are visited, were adapted with an additional decision variable together with the routing decision variables, so that a node visit can be skipped. To that effect, new decision variables are defined as follows.

$$z_{ij} \begin{cases} 1 & \text{if request from loading node } i \in \hat{P} \text{ to unloading node } j \in D^{\star} \text{ is not being satisfied;} \\ 0 & \text{otherwise.} \end{cases}$$

$$z'_{p} \begin{cases} 1 & \text{if no requests at pickup location } p \in P \text{ are being satisfied;} \\ 0 & \text{otherwise.} \end{cases}$$

Decision variables z_{ij} will be applied to constraints relating to the corresponding nodes of the request they refer to. Decision variables z'_p are applied to the constraints relating to the pickup coupling and decoupling nodes and their purpose is to ensure that these nodes are not visited when no requests are being satisfied in that location. These two new sets of variables are linked by constraints (3.46).

$$z'_p \geq \sum_{i \in \hat{P}_p} \sum_{j \in D^{\star}} z_{ij} - \left|\hat{P}_p\right| + 1 \qquad \forall p \in P \tag{3.46}$$

The interpretation of these constraints is as follows: for a given pickup location $p \in P$, if $\sum z_{ij} = \left|\hat{P}_p\right|$, i.e. the number of unaccomplished requests in p is equal to the number of loading nodes, these two summands cancel each other and z'_p will necessarily be equal to 1. Note that it is not necessary to include additional constraints ensuring the contrapositive situation, i.e. that z'_p can only take value 0 if $\sum z_{ij} < \left|\hat{P}_p\right|$, as the model's underlying transportation network is already able to ensure this situation.

The addition of these penalty variables allows the solver to immediately obtain an initial (trivial) solution, where all requests are unaccomplished. Afterwards, the solver is given an initial time limit TL_0 to search for better solutions and start satisfying some requests, so that the matheuristic algorithm that follows does not start so far from the feasible domain.

3.5.2 Solution approach: fix-and-optimise

The proposed solution method consists in an improvement matheuristic under a fix-and-optimise (FO) framework, similar to the one proposed by Helber and Sahling [2010]. The concept behind FO resides in solving smaller MIP sub-problems in an iterative manner so that the computational burden from the high number of integer variables is reduced.

In each iteration, FO uses the best found solution (incumbent solution) as an initial solution for the mathematical solver. From the incumbent solution, a sub-problem is defined by fixing a portion of the binary variables which value is equal to 1, leaving the remaining decision variables (not used in the solution) to be optimised by the mathematical solver.

The outline of the sub-problems is a critical aspect of the FO procedure. Sub-problems need to take into account possible variable interdependencies in order to avoid entrapment in local optima. In our case, variable interdependencies can be verified through vehicles routes dependency. In fact, if two vehicle routes need to be synchronised at a given location and only one of them is fixed, the other route will typically remain the same, as synchronisation constraints will need to be met. Therefore, it is necessary to reach a compromise so that variables dependencies are attenuated and sub-problem size is not significantly large. The outline of the matheuristic is present in Algorithm 3.1. FO sub-problems construction is outlined in lines 7-12. It consists in randomly selecting a given number of pickup locations for which all associated routing decision variables will be released, regardless of the type of vehicle, while fixing all remaining variables in the solution whose value is equal to 1. In the early stages of the algorithm, it is likely that the incumbent solution will be "infeasible" (i.e. having unaccomplished requests), and in these situations, the selection of the first location to be included in the sub-problem will only consider locations whose requests are not yet completely satisfied, thus allowing to reach problem-feasible solutions more quickly.

Algorithm 3.1: Matheuristic outline

input: MIPmodel (mixed integer programming model),
TL_0 (time limit for obtaining an initial solution),
TL_{it} (initial time limit for a solver iteration),
TL_{inc} (time limit increment in solver iteration if new incumbent solution is found),
N (maximum neighbourhood size)

1 x_0, f_0 = `MIPsolve(`MIPmodel, TL_0`)`;
2 $x_{cur} = x_0$; $f_{cur} = f_0$;
3 **while** *termination criteria not met* **do**
4   n = 1; *selected_locs* = ∅;
5   **while** n < N **do**
6   n = n + 1;
7   **release** binary variables of MIPmodel
8   **while** |*selected_locs*| < n **do**
9   **append** random location **to** *selected_locs*
10   **foreach** var **in** x_{cur} **do**
11   **if** var = 1 **and** var **not** *associated with any location* **in** *selected_locs* **then**
12   **fix** var
13   **feed** MIPmodel with initial solution x_{cur}
14   x_{solve}, f_{solve} = `MIPsolve(`MIPmodel, TL_{it}, TL_{inc}`)`;
15   **if** $f_{solve} < f_{cur}$ **then**
16   n = 1; *selected_locs* = ∅; $x_{cur} = x_{solve}$; $f_{cur} = f_{solve}$;
17   **else** n = n + 1;
18 **print** x_{cur}, f_{cur}

The optimisation process of each sub-problem is outlined in line 14. Each optimisation

iteration is given an initial time limit TL_{it}, which is increased by TL_{inc} if a new incumbent solution is found during that initial time.
After obtaining an initial solution and objective function values (x_0, f_0) in lines 1-2, the matheuristic is executed under a VNDS framework where the number of locations in a sub-problem is increased until new incumbent solutions are found. After each FO iteration the resulting solution (x_{solve}, f_{solve}) from the optimisation process is evaluated against the incumbent solution (x_{cur}, f_{cur}) in lines 15-17. In case of an improvement, sub-problem construction is restarted with two random locations ($n = 2$). Otherwise, an additional location is added into the current sub-problem up to the maximum sub-problem size, given by parameter N.

3.6. Computational experiments

The proposed model was applied to a case-study in the biomass supply chain, inspired in a wood chips supplier company operating in Southern Finland. The mathematical model was implemented in the Gurobi 7.5 commercial solver. The solution method was implemented in Python 3.6. The performance of the proposed solution method was tested in problem instances of increasing size in terms of number of pickup/delivery requests and number of vehicles. The results for a baseline instance are described in detail, with emphasis on fleet sizing, the vehicle routes and schedules and the accomplishment of the synchronisation aspects.
The model was also used to compare the obtained plan with the currently adopted planning approach by the supplier, in which synchronisation between lorries and loaders is not taken into account. Finally, some additional managerial insights are provided about the biomass logistics that are relevant to improve planning and decision making in this sector.

3.6.1 Case-study

The company under study is responsible for sizing, assigning and scheduling the vehicles needed for performing the daily chipping and transportation operations that correspond to the contracted wood chips requests. A request is a certain amount of spruce wood chips to be transported between each pickup location and each destination, established beforehand as a result of the biomass distribution plan for that week, as discussed in Marques et al. [2018]. The amounts of material to be transported are expressed in bulk cubic meters (b-m^3), which translate the effective amount of wood chips obtained from raw materials in its "unchipped" state. Empirically, it is commonly accepted that 1 m^{3}of roundwood corresponds to 3 b-m^{3}of wood chips and 1 tonne of wood chips corresponds to 4 b-m^{3}of spruce wood chips [Francescato et al., 2008].
The baseline case for this research is the biomass operations scheduling plan for a certain region "A", encompassing 15 requests of full truck-loads for a given planning day. The capacity of each truck is 76 b-m^{3}and the total amount of wood chips to be delivered is 940.3 b-m^3. There are 5 wood piles of spruce forest residues geographically dispersed and ready to be chipped, also called pickup locations.

The number of available vehicles are 3 lorries, 3 loaders and 7 trucks. Vehicles fixed costs correspond to € 200 for lorries and € 4500 for loaders. The costs related to vehicles travelled distance is 1.00, 1.50 and 1.20 euros per travelled km for lorries, loaders and trucks, respectively. Finally, vehicle costs per expended minute correspond to 0.67, 5.00 and 1.42 euros for lorries, loaders and trucks, respectively.
The considered productivity of loaders is 42 b-m^3/h, meaning that a total of 42 b-m^{3}of wood chips can be produced per hour from the original raw materials [Yoshida et al., 2016]. The service time of each truck-load request is calculated by the quotient of its amount by the loader productivity. A full truck-load will have a loading operation time of 109 min. Given the total available amount of 940.3 b-m^3, the loaders total working time will amount to 22.4h.
Trucks transportation capacity is assumed to be equal to 19 t, which corresponds to a maximum capacity of 76 b-m^3, a value coherent with the estimates given by Francescato et al. [2008]. Each unloading operation at delivery nodes will be proportional to the amount of each transported truck-load so that unloading a truck at its full capacity will amount to 20 min of expended time. Additionally, it is assumed that loader decoupling and coupling operations at depots and pickup locations take 10 min.
The considered duration of the planning horizon is 12 hours, from 8:00 to 20:00. The average travel time between locations is equal to 17.5 min, varying between 3 and 34 min. Travel distances between locations, on the other hand, vary between 2 and 27 km, with an average distance of 13.8 km. The distances between locations were computed by resorting to the national road database of Finland (http://www.liikennevirasto.fi/avoindata/digiroad).
During the data preprocessing phase, the pickup locations were split into truck-load equivalents, as explained in the node creation procedure, which originated the 15 pickup loading nodes corresponding to each request. The average number of requests fulfilled by a pickup location is 3. There are 3 different delivery locations for the sum of the daily requests, meaning that each delivery location, in average, will receive 5 truck-loads. For the baseline instance, a single depot is considered.
Additional instances were considered for analysing the performance of the solution method, as presented in Table 3.3, where instance A5 corresponds to the baseline case described earlier. They consist in variants of the baseline case instance of region "A" through the addition or removal of pickup locations. Specifically, instance A4 corresponds to removing the pickup location that is farther from the depot, instance A3 corresponds to removing the two farthest pickup locations from the depot, while instance A6 corresponds to adding a new pickup location that is closest to the depot, and so on. For the purpose of completeness, a second region "B" is also considered, with different locations and a different fleet. The rationale behind these instances is analogous to the one used for generating instances for region A. For each instance the number of vehicles of each type was progressively reduced or enlarged according to the number of requests demanded in each case. Finally, instances "C" are multi-depot instances, used for the sole purpose of evaluating the adequateness of the solution approach in these situations.
The total quantities to be delivered vary between 535.0 and 1712.3 b-m^3. Due to instances being based on real data, adding or removing one location may result in a big or small

increase or decrease in the total quantity to be transported, depending on the actual quantity the new location has available.

3.6.2 Performance of the solution approaches

The contents of this subsection are threefold: first, the baseline results for the MIP solver approach are introduced and analysed. Afterwards, we evaluate the impact of introducing different combinations of the valid inequalities presented earlier in this paper. Finally, we compare the results obtained by the matheuristic approach with the baseline results.
All computational experiments were performed in an Intel Xeon E5-2680v2 @ 2.80 GHz CPU, with capacity for 20 simultaneous processing threads, with a maximum time limit of two hours for each instance.

3.6.2.1 Exact approach: mixed integer programming solver

Table 3.3 summarizes the obtained results for the MIP solver in all 19 problem instances. In it, the main features of each problem instance are shown, as well as the main indicators of the solution approach, namely the final value of the objective function, the percentual MIP gap computed by Gurobi, the runtime after which no solution improvement was found within the established time limit, as well as the initial and final linear relaxation of the branch-and-bound procedure (root relaxation and best bound, respectively).

Table 3.3: Results for the MIP solver approach

Instance	**Instance Size**								**MIP Solver**							
	Locations			Vehicles available			Requests		Objective Function	MIP Gap	Runtime (s)	Root Relaxation	Best Bound	Model size		Vehicles used (R / L / K)
	$\|O\|$	$\|P\|$	$\|D\|$	$\|R\|$	$\|L\|$	$\|K\|$	No.	Total quantity (b-m3)						Decision variables (bin. / cont.)	Constraints	
A2	1	2	1	2	2	7	9	535.0	**14,386**	12.5%	4,409	9,203	12,584	960 / 179	1,591	1 / 2 / 3
A3	1	3	1	3	3	7	11	682.3	**19,363**	15.5%	4,093	14,321	16,359	1567 / 248	2,368	2 / 2 / 4
A4	1	4	2	3	3	7	13	815.3	**25,549**	31.7%	4,997	16,607	17,449	2110 / 294	3,026	3 / 3 / 4
A5	1	5	3	3	3	7	15	940.3	**27,845**	16.1%	5,144	19,294	23,365	2733 / 340	3,758	3 / 3 / 4
A6	1	6	3	3	3	7	16	974.3	**28,097**	16.0%	4,026	19,910	23,611	3172 / 369	4,245	2 / 3 / 5
A7	1	7	4	4	4	8	18	1,123.3	**41,878**	33.0%	6,982	27,194	28,069	5068 / 504	6,113	3 / 4 / 7
A8	1	8	4	4	4	8	19	1,183.3	**——**	——	——	28,479	29,409	5800 / 540	6,802	——
A9	1	9	4	5	5	9	21	1,279.3	**49,914**	26.7%	6,791	35,290	36,590	8821 / 702	9,296	3 / 4 / 6
B2	1	2	1	2	2	7	8	500.0	**13,131**	4.4%	1,814	8,035	12,557	792 / 163	7,452	2 / 2 / 4
B3	1	3	1	3	3	7	11	720.0	**19,268**	13.0%	420	14,269	16,772	1543 / 248	12,422	2 / 2 / 4
B4	1	4	2	3	3	7	14	900.0	**26,431**	30.6%	2,418	17,954	18,349	2297 / 311	17,210	3 / 3 / 5
B5	1	5	2	3	3	7	15	961.7	**27,256**	13.8%	6,445	19,149	23,496	2711 / 340	1,371	3 / 3 / 5
B6	1	6	2	3	3	7	16	986.7	**28,269**	15.4%	1,822	20,234	23,914	3154 / 369	2,344	3 / 3 / 4
B7	1	7	3	4	4	8	18	1,116.7	**34,840**	21.4%	4,237	26,880	27,371	5044 / 504	3,285	3 / 3 / 5
B8	1	8	3	5	5	9	22	1,386.7	**50,890**	24.7%	5,645	37,186	38,335	8600 / 705	3,740	4 / 4 / 7
B9	1	9	3	6	6	10	26	1,636.7	**——**	——	——	50,576	51,299	13594 / 938	4,227	——
C4	2	4	2	4	4	12	17	1,035.0	**34,243**	20.9%	6,908	27,098	27,098	5788 / 612	6,089	3 / 3 / 6
C6	2	6	2	6	6	12	22	1,402.3	**61,048**	28.1%	7,141	43,899	43,899	11532 / 900	9,413	5 / 4 / 10
C8	2	8	4	6	6	12	27	1,715.3	**——**	——	——	52,834	52,834	16812 / 1098	13,699	——

The results show that Gurobi is unable to prove optimality for any of the solutions if finds within the established time limit, even for smaller instances. For the smaller instance A2 and B2, the estimated MIP gap is 12.5% and 4.4%, respectively, obtained after 4,409 and 2,270 seconds. For the baseline instance A5 the final objective function value with the MIP

solver is obtained after 5144 seconds with a gap of 16.1%. Larger instances (e.g. A8, A9, B8, B9, C6, C8) yield a gap between 20% and 32%. Additional tests for for instances A2, A5 and C6 show that even when the stopping criterion is set to 48 hours the MIP solver still fails to find optimum solution with little improvements in the MIP gap for instances A5 and C6, and the obtained solution for instance A2 is proven to be optimal in approximately 40 hours.
Within the established time limit, Gurobi is able to increase the initial root relaxation in 16 out of the 19 instances, exceptions being instances C4, C6 and C8. In general, the results show that, as instance size increases, the increase in the root relaxation value is less expressive. For example, in instance A2 the root relaxation increases by a order of 3,000, while instance A9 only yields an increase of around 1,000. This analysis suggests that, as instance size increases, Gurobi is not able to keep up with the increase of the number of nodes in the search tree, due to the combinatorial nature of the problem at hand. This conclusion motivates the tests that follow, where several valid inequalities will be tested with the purpose of reducing the size of the branch-and-bound tree in need to be explored by the MIP solver.

3.6.2.2 Exact approach: valid inequalities

In order to validate the adequacy of the valid inequalities in facilitating solution convergence, a new set of computational experiments was run in Gurobi. In these tests, all feasible combinations of valid inequalities were tested for each problem instance, resulting in a total of 15 combinations of valid inequalities and a total of 285 solver runs.
The main indicators used for the assessment of the valid inequalities were the percentual improvements of the root relaxation, best bound and final solution values against the baseline MIP solver approach, without valid inequalities, and within the established time limit. Table 3.4 summarises these indicators.

Table 3.4: Main indicators of the use of valid inequalities

Valid inequalities			**Root Relaxation Improvement**				**Best Bound Improvement**				**Solution Improvement**			
SB1 / SB2 / SB3	**SB4**	**LB1**	Mininum	Median	Average	Maximum	Mininum	Median	Average	Maximum	Mininum	Median	Average	Maximum
—	—	LB1	10.3%	14.5%	14.1%	17.4%	2.5%	11.5%	14.0%	39.9%	-2.1%	0.0%	1.2%	18.4%
	SB4	—	7.9%	11.5%	11.0%	15.2%	4.6%	10.2%	12.7%	39.8%	-1.5%	0.2%	0.7%	7.0%
		LB1	10.3%	14.5%	14.1%	17.4%	4.6%	13.1%	15.8%	42.3%	-3.9%	0.0%	1.2%	17.2%
SB1	—	—	7.9%	11.5%	11.2%	17.5%	2.5%	10.5%	11.5%	21.7%	-13.1%	0.0%	-1.3%	1.2%
		LB1	10.3%	14.5%	14.3%	19.2%	2.0%	12.2%	16.1%	39.9%	-1.7%	0.0%	0.7%	12.2%
	SB4	—	7.9%	11.5%	11.3%	17.5%	4.6%	11.5%	14.0%	39.5%	-3.2%	0.0%	1.5%	19.7%
		LB1	10.3%	14.5%	14.3%	19.2%	4.6%	14.6%	17.4%	40.8%	-13.8%	-0.1%	0.5%	17.3%
SB2	—	—	7.9%	11.5%	11.2%	17.5%	1.5%	9.0%	8.5%	12.2%	-16.0%	-0.3%	-1.6%	1.0%
		LB1	10.3%	14.5%	14.1%	18.3%	1.4%	11.9%	14.4%	40.0%	-19.8%	-0.9%	-2.2%	1.0%
	SB4	—	7.9%	11.5%	11.3%	17.5%	4.6%	9.7%	13.0%	39.2%	-7.9%	-0.1%	-0.8%	1.1%
		LB1	10.3%	14.5%	14.3%	19.2%	4.6%	12.4%	15.1%	42.7%	-18.1%	0.0%	-2.5%	1.3%
SB3	—	—	7.9%	11.5%	11.2%	17.5%	2.1%	9.1%	8.6%	12.2%	-19.2%	-0.9%	-2.7%	1.0%
		LB1	10.3%	14.5%	14.1%	18.3%	3.1%	11.5%	14.2%	39.8%	-20.0%	-1.0%	-2.4%	1.0%
	SB4	—	7.9%	11.5%	11.3%	17.5%	3.9%	9.7%	12.7%	41.3%	-17.4%	-0.4%	-1.8%	1.0%
		LB1	10.3%	14.5%	14.3%	19.2%	4.6%	12.5%	15.0%	41.5%	-17.6%	-0.5%	-2.2%	1.0%

In all combinations of valid inequalities it was possible to obtain an increase of the root relaxation value. The minimum root relaxation percentual increase is 7.9%, and its maximum value is 19.2%. From the different combinations that were tested, the introduction of valid

inequality LB1 into the mathematical model has a significant effect in the root relaxation improvement: in fact, these valid inequalities yield a minimum of 10.3% increase in this indicator. Compared with only using constraints LB1, there is no significant change in the root relaxation improvement when symmetry-breaking constraints are introduced into the model together with constraints LB1, thus suggesting this latter set of constraints absorbs most of the root relaxation improvement.
Unlike the root relaxation improvement, the distribution of the best bound improvement values is far more assimetric. The percentual improvements range from 1.4% up to 42.7%. The variability of this indicator is lowest when using only one of the SB1, SB2 or SB3 types of valid inequalities, although its low variability also determines a significant lower average improvement in the best bound values.
Despite the improvements in the linear relaxation values, this does not necessarily translate into improvements in the final solution: the average improvements in the final solution are mostly negative or around zero percent for each combination of valid inequalities. The results show that the most promising valid inequalities for the problem are the combination of SB1 and SB4 constraints, leading to an average solution improvement of 1.5%. Anyhow, this improvement is observed due to an increase in variability.
In conclusion, these results suggest that the impact of the valid inequalities in the performance of the MIP solver is largely instance-dependent. Therefore, the branch-and-bound procedure reveals itself to be inadequate for these problems, even after strengthening the model with valid inequalities.

3.6.2.3 Matheuristic approach: fix-and-optimise

The used parameters for the matheuristic approach are exhibited in Table 3.5. These parameters were obtained after evaluating the convergence process of the method with different values of these parameters in a preliminary stage. The Gurobi MIP solver was also fed hint values of zero to the penalty variables z_{ij} and z'_p. This feature of Gurobi allows us to inform the solver that these hint values should yield good quality solutions, therefore changing the branching decisions it makes to explore the search tree. All other Gurobi parameters were set to its default values.

Table 3.5: Used parameters for the matheuristic approach

Parameters		**Value**
Termination criteria	Global time limit (TL)	7200s
Obtaining an initial solution	Unit penalty values for z_{pd} variables	10000
	Time limit (TL_0)	300s
Sub-problem construction	Sub-problem sizes	2, 3, 4 pickup locations
	MIP solver iteration time limit (TL_{it})	15s
	Time limit increment in MIP solver iteration if new incumbent is found (TL_{inc})	30s

Table 3.6 summarizes the matheuristic results for the 19 problem instances, including the average value of the best objective function value (bOF) after 10 repetitions for each instance, its standard deviation, the average runtime after which no solution improvement

was found and the number of vehicles used. The percentual difference between the bOF obtained with both methods (% diff.) is also presented.

Table 3.6: Comparison of the MIP solver approach with the matheuristic approach

Instance	**Instance Size**								**MIP Solver**				**Fix-and-Optimise**				**% diff.**
	Locations			Vehicles available			Requests		Objective Function	MIP Gap	Runtime (s)	Vehicles used (R / L / K)	Objective Function		Runtime (s)	Vehicles used (R / L / K)	
	$\|O\|$	$\|P\|$	$\|D\|$	$\|R\|$	$\|L\|$	$\|K\|$	No.	Total quantity (b-m^3)					Average	Std. Deviation			
A2	1	2	1	2	2	7	9	535.0	**14,386**	12.5%	4,409	1 / 2 / 3	**14,386**	**0**	69	1 / 2 / 3	0.0%
A3	1	3	1	3	3	7	11	682.3	**19,363**	15.5%	4,093	2 / 2 / 4	**19,197**	**47**	294	1 / 2 / 3	-0.9%
A4	1	4	2	3	3	7	13	815.3	**25,549**	31.7%	4,997	3 / 3 / 4	**25,461**	**65**	964	2 / 3 / 4	-0.3%
A5	1	5	3	3	3	7	15	940.3	**27,845**	16.1%	5,144	3 / 3 / 4	**27,294**	**89**	1,290	2 / 3 / 5	-2.0%
A6	1	6	3	3	3	7	16	974.3	**28,097**	16.0%	4,026	2 / 3 / 5	**27,754**	**108**	2,071	3 / 3 / 5	-1.2%
A7	1	7	4	4	4	8	18	1,123.3	**41,878**	33.0%	6,982	3 / 4 / 7	**34,850**	**138**	3,474	3 / 3 / 5	-16.8%
A8	1	8	4	4	4	8	19	1,183.3	——	——	——	——	**41,761**	**179**	4,081	4 / 4 / 6	——
A9	1	9	4	5	5	9	21	1,279.3	**49,914**	26.7%	6,791	3 / 4 / 6	**49,294**	**97**	4,412	4 / 4 / 6	-1.2%
B2	1	2	1	2	2	7	8	500.0	**13,131**	4.4%	1,814	2 / 2 / 4	**13,131**	**0**	47	2 / 2 / 4	0.0%
B3	1	3	1	3	3	7	11	720.0	**19,268**	13.0%	420	2 / 2 / 4	**19,295**	**50**	248	2 / 2 / 4	0.1%
B4	1	4	2	3	3	7	14	900.0	**26,431**	30.6%	2,418	3 / 3 / 5	**26,243**	**53**	775	3 / 3 / 5	-0.7%
B5	1	5	2	3	3	7	15	961.7	**27,256**	13.8%	6,445	3 / 3 / 5	**27,237**	**97**	1,437	3 / 3 / 5	-0.1%
B6	1	6	2	3	3	7	16	986.7	**28,269**	15.4%	1,822	3 / 3 / 4	**27,734**	**91**	1,770	3 / 3 / 5	-1.9%
B7	1	7	3	4	4	8	18	1,116.7	**34,840**	21.4%	4,237	3 / 3 / 5	**34,410**	**104**	2,910	2 / 3 / 5	-1.2%
B8	1	8	3	5	5	9	22	1,386.7	**50,890**	24.7%	5,645	4 / 4 / 7	**49,970**	**119**	5,054	3 / 4 / 7	-1.8%
B9	1	9	3	6	6	10	26	1,636.7	——	——	——	——	**69,323**	**266**	4,860	5 / 5 / 8	——
C4	2	4	2	4	4	12	17	1,035.0	**34,243**	20.9%	6,908	3 / 3 / 6	**33,904**	**150**	1,452	3 / 3 / 5	-1.0%
C6	2	6	2	6	6	12	22	1,402.3	**61,048**	28.1%	7,141	5 / 4 / 10	**56,522**	**79**	2,392	3 / 4 / 7	-7.4%
C8	2	8	4	6	6	12	27	1,715.3	——	——	——	——	**70,273**	**197**	5,350	4 / 5 / 8	——
																Average	**-2.3%**

For the matheuristic approach, small values of standard deviation were found for all problem instances, thus suggesting that 10 repetitions per instance are sufficient for obtaining representative results. As the number of pickup locations increases, as well as the number of truck-loads to be transported, the runtime is higher, as expected. For small instances (e.g. A2, B2), it takes around one minute to find good admissible solutions. For instance A5 the bOF is reached after 22min and for larger instances it takes close to 1h30min.

Within the two hours time limit, the average bOF is within the same range of the MIP solver for practically all problem instances and it is obtained in much lower computational times. However, in the majority of the cases (12 in 16) the difference of the bOF in both methods is below 2%. The major differences are found in instances A7 (yielding a bOF 16.9% better in FO than in the MIP solver) and in C6. In these cases, there is a significant difference in the total number of vehicles used. For example, for instance A7, FO proposes 3 lorries, 3 loaders and 5 trucks while the MIP solver estimates one additional loader and two additional trucks. In the case of instance C6, FO proposes 4 lorries, 4 loaders and 7 trucks, while the MIP solver allocates one more lorry and three more trucks. In the other problem instances, both solution methods point to equal conclusions regarding the number of vehicles used, therefore the difference in terms of bOF are due to distinct scheduling and synchronisation solutions.

In conclusion, the FO solution approach seems to be adequate to solve this problem. As the number of pickup locations increases, it provides good quality solutions in shorter computational time when compared with the MIP solver. For instances above 8 pickup locations the MIP solver is typically unable to even obtain an initial solution, while the matheuristic is able to obtain admissible solutions within the two hours time limit.

3.6.3 Baseline instance solution analysis

The results for the baseline instance A5, obtained with the FO approach, show that all daily requests can be fulfilled with a fleet composed by 2 lorries, 3 loaders and 5 trucks. The total transportation costs amount to € 27,301, from which 51% (€ 13,900) correspond to daily usage costs (i.e. fixed costs), 3% (€ 799) are travel costs and 46% to vehicle time costs. Almost all fixed costs are related to the daily use of the loaders (€ 13,500). Lorries correspond to approximately 3% of the daily usage costs. In respect to travel costs, € 493 are related with the routes of the trucks, while € 306 relate with the routes of loaders and lorries.

As expected, the loaders are the bottleneck equipment of biomass logistics. In total, the loaders remain idle only 61 min out of the used 1708 min of the shift, yielding an average of 20 min per loader. The loader remains at a pickup location 5 h in average. Consequently, lorries are sub-used, as its average idle time is 3h12min per lorry.

It is noteworthy that the model indeed captures all the business aspects that lead to the definition of feasible routes in this complex real-world problem. The routes of lorries r_1 and r_2 are showed in Figure 3.5a, based on the real-life coordinates of all locations (depots, pickup locations and delivery locations). As an example, r_1 departures from depot o_1 carrying l_1 to unload it in pickup location p_3. Then, it travels empty back to depot o_1, from where it departs with loader l_3 to pickup location p_4. Next, it travels back empty to depot o_1 and finishes the route. Note that all pickup locations are visited by lorries twice, both for loader drop-off and pickup, which suggests that it is more advantageous to have lorries perform multiple loader pickups and drop-offs instead of pairing with a single loader and wait in the pickup location until loading operations are done. This is a consequence of the value of the lorry daily usage cost and the ratio of the latter and the variable transportation cost, as discussed in section 3.6.5.

Figure 3.5a also presents the pickup assignments for loaders l_1, l_2 and l_3. In this case, l_2 is assigned to the farthest pickup location, p_5, with 5 truck-loads, so the schedule of l_2 consists in departing from the depot in the beginning of the day heading to pickup location p_5 (arrival at 8:18), remaining there all day and returning to the depot in the end of the day at 19:07. Conversely, l_1 and l_3 are assigned to pickup locations closer to depot o_1 and with less truck-loads, being able to visit more than one location during the day. In fact, loader l_1 is the mostly used resource during the day.

The routes and schedules for the trucks are also presented in Figure 3.5a. Pickup location p_5 receives 4 out of the 5 available trucks and this can be explained due to the fact that operations at p_5 have a very little time slack, and therefore the location requires trucks to arrive quickly so that the minimum amount of time is spent. After delivering truck-loads coming from p_5 to delivery location d_1, trucks usually visit other pickup locations with less truck-loads, with preference for the ones that do not lead to queueing at the location.

The timeline with the sequence of operations in pickup location p_2 is showed in Figure 3.5b, evidencing the synchronisation aspects that were correctly dealt by the model. Loader l_1 is dropped at location p_2 by lorry r_2 at 11:45. Loader drop-off takes 10 min and then stays idle for 21 min until the first truck (k_7) arrives at 12:11. The synchronised operations are wood chipping (by l_1) and truck container loading (by k_7), and takes about 66 min. The second truck (k_2) arrives just 2 min after the departure of k_7 and synchronised operations are immediately started. After the second truck-load is served, the availability in p_2 is exhausted and the loader is now ready for transport. Lorry r_2 returns to location p_2 at 13:41 and waits until 15:08 for the loader to be available for pickup. Movement synchronisation corresponds to both the arrival of r_2 with l_1 in the morning of the day and the departure of r_2 with l_1 in the end of the day.

3.6.4 Comparison with current planning process

The biomass operational planning process of our case-study consisted in elaborating the operational plans manually. This was due to the fact that no planning tool for integrated fleet sizing and vehicle routing that acknowledged inter-vehicle synchronisation was available. To that effect, the elaboration of plans consisted in decomposing the decisions into a fleet sizing phase, followed by a route planning phase.

One of the main limitations of this planning process consisted in the fact that not all synchronisation aspects were considered because it would turn route construction more difficult to address manually. In fact, lorries and loaders were assumed as being unique vehicles, and therefore eventual gains from a lorry performing multiple loader pickups and drop-offs were not considered.

The fleet sizing phase relies on empirically-driven rules, such as:

1. the number of loaders required cannot exceed the number of pickup locations, and can be estimated as a function of the total amount of material available for chipping at each pickup location, a_p, the loader average available time for chipping operations, ATL, and the loader average productivity, w (equation (3.47)). This function resembles the OEE (Overall Equipment Effectiveness) performance indicator that is widely used in the TPM (Total Productive Maintenance) methodology (e.g., Singh et al. [2013]);

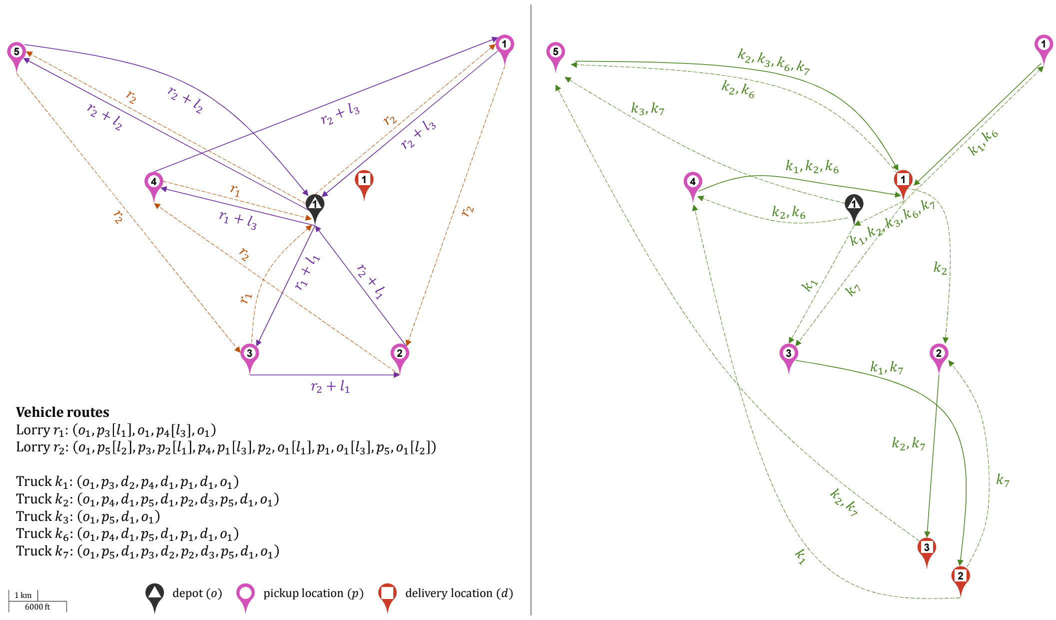
$$|L| = \frac{\sum_{p \in P} a_p}{ATL \cdot w} \tag{3.47}$$

2. the number of lorries is equal to the number of loaders. As already stated above, the possibility of decoupling the lorry and the loader was not considered, and therefore lorries and loaders will always move simultaneously;

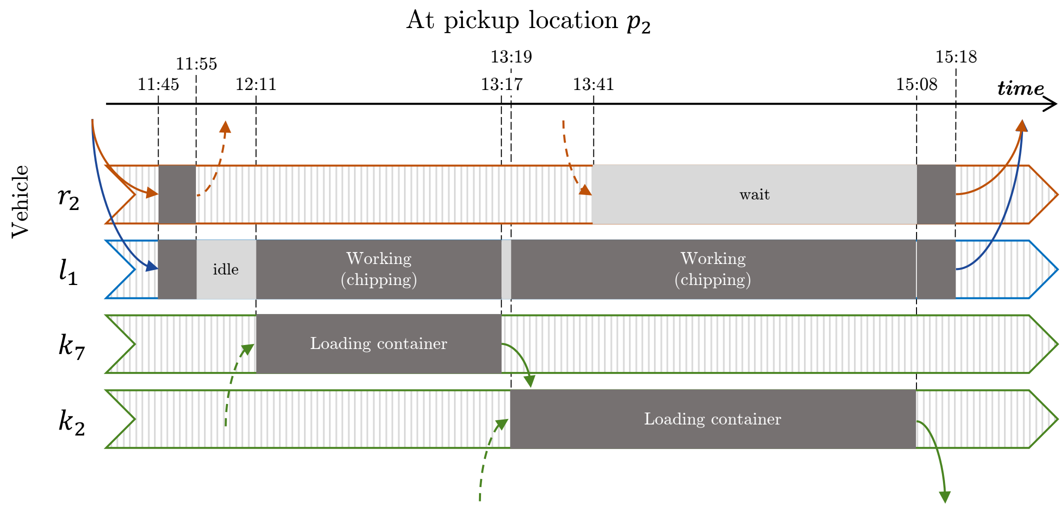
3. the number of trucks required can be estimated as a function of the total amount of wood available for chipping at each pickup location, a_p, the trucks capacity, Q, the truck average available time for transportation (ATK), the average travel time between locations, TT, and the average loading and unloading time, TO (equation (3.48));

$$|K| = \frac{\sum_{p \in P} a_p \cdot 2(TT + TO)}{Q \cdot ATK} \tag{3.48}$$

In this study, the total amount of available material is 940.3 b-m^3. The average loader



(a) Map with vehicles routes represented



(b) Operations timeline for pickup location p_2

Figure 3.5: Obtained solution for the baseline instance

available time is estimated in 6 h based on expert's opinions, corresponding to the amount of time (in the daily shift) that the loader can be working, subtracting all stopping time (i.e. transportation to and from pickup locations, breaks, faults). The average trucks available time is estimated in 7h, based on expert's opinions, corresponding to the amount of time (in the daily shift) that the truck and driver can be working, subtracting all stopping time (i.e. transportation to and from the depot, breaks).

According to this approach the resulting number of loaders required is 3.73, rounded up to 4, therefore one more loader than in the plan obtained with the proposed solution approach. The number of lorries required is also equal to 4, which is double the number of lorries proposed by our solution approach. In respect to trucks, the number of lorries required is 7.4, rounded up to 8, therefore 2 more trucks needed than in the solution obtained in the matheuristic approach.

In order to obtain the routes corresponding to the previously sized fleet, the matheuristic approach was run again assuming the usage of this fleet and considering that lorries and loaders cannot be split. For this purpose the mathematical formulation presented in Section 3.4 was adapted to include constraints that enforce the use of these vehicles and the transportation network preprocessing was also changed, as explained in 3.4.1.4.

The obtained costs increased to € 38,142, which are 40% higher than the previously obtained plan. In fact, because the number of lorries (and loaders) is almost equal to the number of pickup locations, each lorry and loader will visit only one pickup location except for one loader and one lorry, which will visit two. Therefore, a lorry and loader visit pickup location p_1, followed by location p_2 after p_1 is completed. Due to truck-loads of these locations being smaller than the ones in the remaining locations, the plan chooses these two locations to be performed sequentially rather than simultaneously, although their relative distance is fairly large.

Table 3.7 shows the main indicators for this current plan. Compared with our baseline plan, we are able to acknowledge the significant increase in vehicles fixed costs, while travel costs remain stable. Vehicles idle times, however, have greatly increased for lorries and trucks. Lorries idle time has increased due to these vehicles now having to wait for loaders until they have completed all truck-loads. Trucks idle time increase can be explained by the fact that most pickup locations are being processed in an almost simultaneous manner, leading to a high demand of trucks at pickup locations practically at the same time.

From this analysis it is possible to conclude that the routing of these vehicles is simplistic when compared with the proposed plan, as such a solution will incur into unnecessary vehicle fixed costs when tasks could be concentrated in a lower number of vehicles, thus taking advantage of the length of the planning horizon. These results therefore suggest that the current planning approach did not yield satisfactory plans because it used a significantly higher number of vehicles than necessary, thus increasing fixed costs without having significant gains in vehicles idle time or travel costs that would compensate its usage in the first place.

3.6.5 Managerial insights for biomass logistics planning

The proposed matheuristic approach was further tested for providing additional managerial insights about the impact of key parameters of biomass logistics systems over the optimal plan. For this purpose, two main parameters were selected based on experts opinions. Alternative scenarios were built for the baseline instance by changing the parameters values within a range of possible values taken from the literature and model results were compared.

The first selected parameter is the loader's productivity, as it defines the time needed to perform the loading operations at pickup locations and therefore significantly impacts the ability of a lorry having sufficient time to perform multiple loader pickups and drop-offs during the route. The predicted impacts can range from a lower number of used lorries and/or loaders to lower vehicle idle times in general. The considered values of loaders productivity were from 36, 42 and 60 b-m^3/h, values that are coherent with the ones in Francescato et al. [2008]. The second selected parameter is the lorries fixed costs, as it is believed that the ratio between this parameter and its variable costs can significantly impact the number of used lorries. The considered fixed costs were € 0, € 100 and € 200.

Consequently, 8 alternative scenarios were generated with the combination of the possible parameter values. The results are presented in Table 3.7.

Table 3.7: Main information about the different studied scenarios

Scenario	Productivity (b-m^3/h)	Lorries fixed cost (EUR)	Runtime (s)	Vehicles used (*R* / *L* / *K*)	Objective Function (EUR)	Fixed costs (EUR)	Travel costs (EUR)	Total idle time (min)			Decoupling ratio
								Lorries	Loaders	Trucks	
Baseline	**42**	**200**	**1,804**	**2 / 3 / 5**	**27,301**	**13,900**	**799**	**383**	**61**	**320**	**100%**
Current plan	**42**	**200**	**304**	**4 / 4 / 8**	**38,142**	**18,800**	**739**	**1,458**	**65**	**519**	**0%***
Scenario 1	42	100	6,502	3 / 3 / 5	26,509	13,800	803	307	5	384	100%
Scenario 2	42	0	1,546	3 / 3 / 5	26,226	13,500	804	307	8	380	100%
Scenario 3	36	200	1,707	3 / 3 / 5	28,179	14,100	812	394	15	256	100%
Scenario 4	36	100	983	3 / 3 / 5	28,020	13,800	814	408	14	348	100%
Scenario 5	36	0	526	3 / 3 / 5	27,929	13,500	798	408	14	519	100%
Scenario 6	60	200	907	2 / 2 / 4	19,296	9,400	741	363	19	346	0%
Scenario 7	60	100	1,789	2 / 2 / 4	19,183	9,200	695	548	1	450	80%
Scenario 8	60	0	765	2 / 2 / 4	19,096	9,000	719	549	5	454	80%

Decoupling ratio = No. of decoupling situations / No. total pickup locations; * - by definition.

The comparison of the results of the alternative scenarios shows a clear trend for using less vehicles when loaders are more productive. The decrease in the number of vehicles reflects itself not only on the number of used loaders, but also on the number of used lorries and trucks. This is due to the fact that the usage of less loaders implies the occurrence of less simultaneous loading operations, which in turn reduces the need for additional lorries and trucks.

The increase of the loader productivity from 36 b-m^3/h to 42 b-m^3/h appears to have very little impact in the obtained solution, as it does not change the number of used vehicles and the decrease of the objective function is approximately 5%. However, increasing the loader productivity from 42 b-m^3/h to 60 b-m^3/h has a very significant impact, mainly due to the decrease in vehicles fixed costs.

Results analysis also contains a decoupling ratio indicator, which tells the proportion of

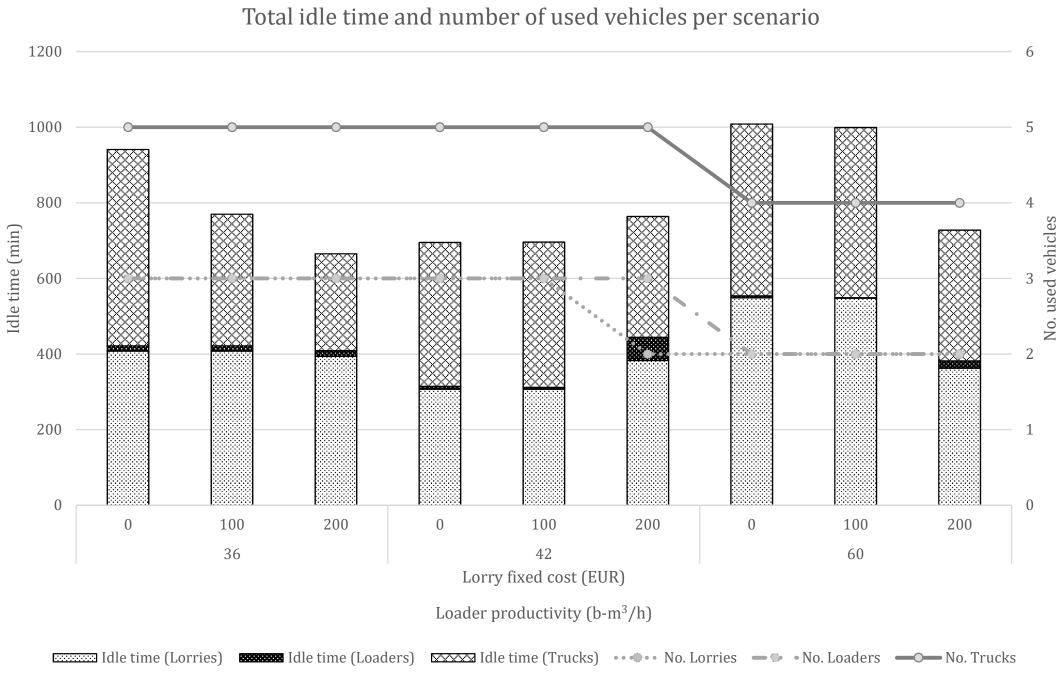


Figure 3.6: Main information about the different studied scenarios

occurrences where a lorry was decoupled from the loader to perform different routes. In almost all the scenarios, including the baseline, the decoupling ratio is equal to 100%, meaning that for all the five pickup locations the optimal route plans encompasses decoupling the lorry and loader, i.e. the lorry departures just after the loader drop-off and the same or another lorry comes back when the loader is ready for transport. However, when the loader productivity is equal to 60 b-m^3/h and lorries fixed cost is € 200 (scenario 6), the decoupling ratio is equal to zero, meaning that no loader decouplings were made and all lorries and loaders behaved as unique vehicles.

In respect to the analysis of the vehicles idle time, represented in Figure 3.6, results suggest that when the loader productivity is increased to 42 b-m^3/h the total idle time also increases to approximately 1,000 min, specially due to the increase of the idle time of the lorries at the pickup locations. For the scenarios with lower productivity the total idle time is between 600 and 800 minutes, similar to the baseline.

Contrarily to our expectations, decreasing the lorry fixed cost has little impact in the increase of the total distance costs, i.e. it does not stimulate decoupling. The results also suggest that performing loader decouplings at pickup locations is an advantageous choice for low loader productivity values. For high productivity values there can be an incentive for not performing decouplings, namely the lorry fixed costs. For the studied scenarios, a lorry fixed cost of € 200 with a productivity of 60 b-m^3/h induces lorries to be used as efficiently as possible, which in this case is achieved by waiting for its loader to finish operations at the location, as loading times are now shorter.

From the obtained results it is possible to conclude that, for this instance, the main factor favouring loader decoupling is its productivity and not the variation of lorry fixed costs.

3.7. Conclusions and future work

In this work, a full truck-load pickup and delivery problem with multiple vehicle synchronisation is presented, motivated by a real-world case-study in the biomass supply chain in Finland. Although the literature about pickup and delivery problems is vast, there is a gap in the literature considering the synchronisation of different types of vehicles with the purpose of avoiding unproductive times while at the same time successfully modelling operations and movement synchronisation of lorries and trailers.

A novel modelling approach was developed in a systematic manner, where synchronisation aspects were dealt in the model preprocessing stage, in the definition of the decision variables and in specific time synchronisation constraints. With the purpose of tightening the mathematical formulation, a set of valid inequalities was also devised. The computational experiments performed with these valid inequalities allowed to conclude that this approach was unable to facilitate convergence into greater quality solutions.

Due to all the vehicle route dependencies and a highly pre-processed transportation network, a matheuristic approach based on the fix-and-optimise principles was developed, which not only proved to obtain better results than a traditional MIP solver but also was able to obtain feasible solutions for more constrained problem instances while the MIP solver was unable to find an initial solution.

The direct impact of this model and solution approach yields significant cost savings when compared with the current planning approach, which was a cumbersome task, performed manually. The cost savings consist in a higher utilisation ratio of each vehicle by considering that lorries can be decoupled from loaders and may perform multiple loader pickups and drop-offs, therefore decreasing their unproductive times. This increase in vehicle efficiency allows for a significant reduction of the number of necessary vehicles to perform the scheduled operations, and therefore avoids incurring into unnecessary vehicle fixed costs.

Additional tests were made with the purpose of studying the impact of two main instance parameters, namely loader productivity, which defines the average duration of each loading task at pickup locations, and the lorry fixed cost. The performed analysis suggests that a higher loader productivity can greatly impact the structure of the routing plan, as lorries and loaders will tend to behave as a single vehicle.

These last results may be interesting to consider in future work. More general models may be developed where loader productivity may be considered variable according to the location and the vehicle itself, which would allow to account for trade-offs between performing loading operations faster with more expensive loaders or performing loading operations slower with lower fixed costs. Future work may also consider alternative models for tackling multiple vehicle synchronisation under different modelling perspectives, the introduction of uncertainty factors in the problem through robust or stochastic optimisation or the development of dynamic routing approaches.

Acknowledgements

This research was partly supported by the PhD grant SFRH/BD/136314/2018, awarded by the Portuguese Foundation for Science and Technology (FCT), and by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e Tecnologia within project POCI-01-0145-FEDER-016733. Further funding was received from the Portugal 2020 project "DM4Manufacturing - Aligning manufacturing decision making with advanced manufacturing technologies", POCI-01-0145-FEDER-016418, financed by UE/FEDER through the program COMPETE2020. This support is gratefully acknowledged.

Bibliography

Yossiri Adulyasak, Jean-François Cordeau, and Raf Jans. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS Journal on Computing*, 26(1):103–120, feb 2014. doi: 10.1287/ijoc.2013.0550.

J. Allen, M. Browne, A. Hunter, J. Boyd, and H. Palmer. Logistics management and costs of biomass fuel supply. *International Journal of Physical Distribution & Logistics Management*, 28(6):463–477, 1998. doi: 10.1108/09600039810245120.

Tasseda Boukherroub, Luc LeBel, and Sébastien Lemieux. An integrated wood pellet supply chain development: Selecting among feedstock sources and a range of operating scales. *Applied Energy*, 198:385–400, July 2017. doi: 10.1016/j.apenergy.2016.12.013.

David Bredström and Mikael Rönnqvist. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, 191(1):19–31, 2008. ISSN 0377-2217. doi: 10.1016/j.ejor.2007.07.033.

I-Ming Chao. A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29(1):33–51, 2002. ISSN 0305-0548. doi: 10.1016/S0305-0548(00)00056-3.

J-F Cordeau, G Laporte, and A Mercier. A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52(8):928–936, August 2001. doi: 10.1057/palgrave.jors.2601163.

G. B. Dantzig and J. H. Ramser. The Truck Dispatching Problem. *Management Science*, 6 (1):80–91, 1959. doi: 10.1287/mnsc.6.1.80.

Ulrich Derigs, Markus Pullmann, and Ulrich Vogel. Truck and trailer routing—Problems, heuristics and computational experience. *Computers & Operations Research*, 40(2): 536–546, 2013. ISSN 0305-0548. doi: 10.1016/j.cor.2012.08.007.

Michael Drexl. *On Some Generalized Routing Problems*. Ph.D. Thesis, Deutsche Post Lehrstuhl für Optimierung von Distributionsnetzwerken (NN), 2007.

Michael Drexl. Branch-and-Price and Heuristic Column Generation for the Generalized Truck-and-Trailer Routing Problem. *Journal of Quantitative Methods for Economics and Business Administration*, 12(0):5–38, 2011.

Michael Drexl. Synchronization in Vehicle Routing—A Survey of VRPs with Multiple Synchronization Constraints. *Transportation Science*, 46(3):297–316, 2012. doi: 10.1287/trsc.1110.0400.

Patrik Flisberg, Bertil Lidén, and Mikael Rönnqvist. A hybrid method based on linear programming and tabu search for routing of logging trucks. *Computers & Operations Research*, 36(4):1122–1144, April 2009. doi: 10.1016/j.cor.2007.12.012.

Valter Francescato, Eliseo Antonini, and Luca Zuccoli Bergomi. Wood Fuels Handbook. 2008.

Michel Gendreau, François Guertin, Jean-Yves Potvin, and Éric Taillard. Parallel Tabu Search for Real-Time Vehicle Routing and Dispatching. *Transportation Science*, 33(4): 381–390, November 1999. doi: 10.1287/trsc.33.4.381.

Veaceslav Ghilas, Emrah Demir, and Tom Van Woensel. An adaptive large neighborhood search heuristic for the Pickup and Delivery Problem with Time Windows and Scheduled Lines. *Computers & Operations Research*, 72:12–30, August 2016. doi: 10.1016/j.cor.2016.01.018.

Axel Grimault, Nathalie Bostel, and Fabien Lehuédé. An adaptive large neighborhood search for the full truckload pickup and delivery problem with resource synchronization. *Computers & Operations Research*, 88:1–14, December 2017. ISSN 0305-0548. doi: 10.1016/j.cor.2017.06.012.

Manfred Gronalt and Patrick Hirsch. Log-Truck Scheduling with a Tabu Search Strategy. In Karl F. Doerner, Michel Gendreau, Peter Greistorfer, Walter Gutjahr, Richard F. Hartl, and Marc Reimann, editors, *Metaheuristics: Progress in Complex Systems Optimization*, pages 65–88. Springer US, Boston, MA, 2007. ISBN 978-0-387-71921-4. doi: 10.1007/978-0-387-71921-4_4.

Timo Gschwind, Stefan Irnich, Ann-Kathrin Rothenbächer, and Christian Tilk. Bidirectional labeling in column-generation algorithms for pickup-and-delivery problems. *European Journal of Operational Research*, 266(2):521–530, April 2018. doi: 10.1016/j.ejor.2017.09.035.

Nizar El Hachemi, Michel Gendreau, and L.-M. Rousseau. A Heuristic to Solve the Weekly Log-Truck Scheduling Problem. In *International Conference on Industrial Engineering and Systems Management*, May 2009. ISBN 978-2-9600532-2-7.

Stefan Helber and Florian Sahling. A fix-and-optimize approach for the multi-level capacitated lot sizing problem. *International Journal of Production Economics*, 123(2): 247–256, February 2010. ISSN 0925-5273. doi: 10.1016/j.ijpe.2009.08.022.

Patrick Hirsch. Minimizing Empty Truck Loads in Round Timber Transport with Tabu Search Strategies. *International Journal of Information Systems and Supply Chain Management*, 4(2):15–41, April 2011. doi: 10.4018/jisscm.2011040102.

Byung-In Kim, Jeongin Koo, and Junhyuk Park. The combined manpower-vehicle routing problem for multi-staged services. *Expert Systems with Applications*, 37(12):8424–8431, December 2010. ISSN 0957-4174. doi: 10.1016/j.eswa.2010.05.036.

Jarl Eirik Korsvik, Kjetil Fagerholt, and Gilbert Laporte. A large neighbourhood search heuristic for ship routing and scheduling with split loads. *Computers & Operations Research*, 38(2):474–483, February 2011. doi: 10.1016/j.cor.2010.07.005.

Fabien Lehuédé, Philippe Grangier, Axel Grimault, and Renaud Masson. Efficient handling of synchronization in three vehicle routing problems. In *Odysseus 2015*, Odyss, Ajaccio, France, May 2015.

Shih-Wei Lin, Vincent F. Yu, and Shuo-Yan Chou. Solving the truck and trailer routing problem based on a simulated annealing heuristic. *Computers & Operations Research*, 36(5):1683–1692, 2009. ISSN 0305-0548. doi: 10.1016/j.cor.2008.04.005. Selected papers presented at the Tenth International Symposium on Locational Decisions (ISOLDE X).

Dorota Slawa Mankowska, Frank Meisel, and Christian Bierwirth. The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, 17(1):15–30, March 2014. ISSN 1572-9389. doi: 10.1007/s10729-013-9243-1.

Alexandra Marques, Jussi Rasinmäki, Ricardo Soares, and Pedro Amorim. Planning woody biomass supply in hot systems under variable chips energy content. *Biomass and Bioenergy*, 108:265–277, 2018. ISSN 0961-9534. doi: 10.1016/j.biombioe.2017.11.016.

Frank Meisel and Herbert Kopfer. Synchronized routing of active and passive means of transport. *OR Spectrum*, 36(2):297–322, dec 2012. doi: 10.1007/s00291-012-0310-7.

Annelies De Meyer, Dirk Cattrysse, and Jos Van Orshoven. A generic mathematical model to optimise strategic and tactical decisions in biomass-based supply chains (OPTIMASS). *European Journal of Operational Research*, 245(1):247–264, August 2015. doi: 10.1016/j.ejor.2015.02.045.

F. Neves-Moreira, P. Amorim, L. Guimarães, and B. Almada-Lobo. A long-haul freight transportation problem: Synchronizing resources to deliver requests passing through multiple transshipment locations. *European Journal of Operational Research*, 248(2): 487–506, 2016. ISSN 0377-2217. doi: 10.1016/j.ejor.2015.07.046.

M. Palmgren, M. Ronnqvist, and P. Varbrand. A near-exact method for solving the log-truck scheduling problem. *International Transactions in Operational Research*, 11(4): 447–464, July 2004. doi: 10.1111/j.1475-3995.2004.00469.x.

Sophie N. Parragh and Jean-François Cordeau. Branch-and-price and adaptive large neighborhood search for the truck and trailer routing problem with time windows. *Computers & Operations Research*, 83:28–44, 2017. ISSN 0305-0548. doi: 10.1016/j.cor.2017.01.020.

Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(1):21–51, April 2008a. ISSN 1614-631X. doi: 10.1007/s11301-008-0033-7.

Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(2):81–117, June 2008b. ISSN 1614-631X. doi: 10.1007/s11301-008-0036-4.

Hanne L. Petersen and Stefan Ropke. The Pickup and Delivery Problem with Cross-Docking Opportunity. In *Lecture Notes in Computer Science*, pages 101–113. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-24264-9_8.

David Pisinger and Stefan Ropke. Large Neighborhood Search. In *Handbook of Metaheuristics*, pages 399–419. Springer US, 2010. doi: 10.1007/978-1-4419-1665-5_13.

Stefan Ropke and David Pisinger. An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transportation Science*, 40(4): 455–472, 2006. doi: 10.1287/trsc.1050.0135.

Stefan Ropke, Jean-François Cordeau, and Gilbert Laporte. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, 49(4):258–272, 2007. doi: 10.1002/net.20177.

Stephan Scheuerer. A tabu search heuristic for the truck and trailer routing problem. *Computers & Operations Research*, 33(4):894–909, 2006. ISSN 0305-0548. doi: 10.1016/j.cor.2004.08.002. Part Special Issue: Optimization Days 2003.

Hanif D. Sherali and J. Cole Smith. Improving discrete model representations via symmetry considerations. *Management Science*, 47(10):1396–1407, oct 2001. doi: 10.1287/mnsc.47.10.1396.10265.

Ranteshwar Singh, Dhaval B. Shah, Ashish M. Gohil, and Milesh H. Shah. Overall Equipment Effectiveness (OEE) Calculation - Automation through Hardware & Software Development. *Procedia Engineering*, 51:579–584, 2013. doi: 10.1016/j.proeng.2013.01.082.

Christian Tilk, Nicola Bianchessi, Michael Drexl, Stefan Irnich, and Frank Meisel. Branch-and-price-and-cut for the active-passive vehicle-routing problem. *Transportation Science*, 52(2):300–319, mar 2018. doi: 10.1287/trsc.2016.0730.

Marjolein Veenstra, Marilène Cherkesly, Guy Desaulniers, and Gilbert Laporte. The pickup and delivery problem with time windows and handling operations. *Computers & Operations Research*, 77:127–140, January 2017. doi: 10.1016/j.cor.2016.07.014.

Mika Yoshida, Simon Berg, Rin Sakurai, and Hideo Sakai. Evaluation of Chipping Productivity with Five Different Mobile Chippers at Different Forest Sites by a Stochastic Model. *Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering*, 37(2):309–318, June 2016.

CHAPTER 4

Dealing with uncertainty in routing problems with synchronisation

---

A robust optimisation approach for the vehicle routing problem with synchronisation

Ricardo Soares [*†] · **Sophie N. Parragh**[‡] · **Pedro Amorim**[*†] · **Alexandra Marques**[*]

Working paper

Abstract The Vehicle Routing Problem (VRP) with Synchronisation aims to minimise the total routing costs while considering synchronisation requirements that must be fulfilled between tasks of different routes. Although several works in the literature have tackled this problem extensively, mainly the deterministic version has been tackled so far. This paper presents a robust optimisation approach for the VRP with Synchronisation taking into consideration the underlying uncertainty in vehicle travel times between customers. This work leverages existing approaches in the literature to envisage two mathematical models for the Robust VRP with Synchronisation, as well as a branch-and-cut algorithm to solve more difficult problem instances. A set of computational experiments is also devised and presented to obtain insights regarding key performance parameters of the mathematical models and the solution algorithm.

Keywords vehicle routing · robust optimisation · synchronisation · branch-and-cut

[*]INESC TEC – Institute for Systems and Computer Engineering, Technology and Science, Porto, Portugal
[†]FEUP – Faculty of Engineering, University of Porto, Porto, Portugal
[‡]Institute of Production and Logistics Management / JKU Business School, Johannes Kepler University Linz, Linz, Austria

4.1. Introduction

The Vehicle Routing Problem with Synchronisation is an extension of the standard Vehicle Routing Problem (VRP) where the feasibility of a route may be dependent on the feasibility of another route. This definition covers problems where different vehicles need to visit customers within given time offsets – as it happens with certain logistic operations requiring more than one vehicle for them to be completed –, or when different vehicles need to be travelling simultaneously – as it happens with problems using tractors and trailers, for example.

Due to the specificities of the VRP with Synchronisation compared to traditional VRPs, this problem variant poses an additional challenge: the fact that each route of a given solution cannot be evaluated independently complicates not only the evaluation of the feasibility of a given solution but also hinders the exploration of the solution space.

Probably due to this additional complexity, research work on the VRP with Synchronisation has primarily focused on deterministic problems. Very few examples of VRPs with Synchronisation subject to uncertainty can be found in the literature. Furian et al. [2018] consider the home health care routing case with operation synchronisation and propose an optimisation procedure embedded in a discrete-event simulation framework which handles stochasticity. Stochasticity in VRPs with Synchronisation has already been identified as being a research gap worth being addressed as this problem variant turns more mature [e.g., Parragh and Doerner, 2018].

Presenting a suitable approach for a non-deterministic VRP with Synchronisation is especially challenging because transitioning from a deterministic to a non-deterministic setting usually entails significant overhead in computational efficiency. This, allied to the fact that these problems are inherently complex, allows us to infer that VRPs with Synchronisation subject to uncertainty should be highly intractable.

Nevertheless, the relevance of acknowledging uncertainty in such problems should not be neglected. In fact, due to its underlying route interdependence, uncertainty in VRPs with Synchronisation can completely change the implementation of a routing plan, given that delays in performing specific tasks can propagate to other tasks and, in a worst-case scenario, can render the entire plan infeasible.

In the literature, stochastic programming and robust optimisation constitute the two main approaches for dealing with uncertainty in VRPs [Averbakh, 2001]. In stochastic programming, knowing the probability distributions of the uncertain parameters is usually required, or at least being able to estimate them. Its goal consists of optimising a solution's expected value while retaining solution feasibility for a variety of alternative scenarios [Birge and Louveaux, 2011]. On the other hand, robust optimisation does not require that probability distributions be known. This approach assumes that the values taken by the uncertain parameters are contained within an uncertainty set. The optimal solution must, therefore, be feasible for all possible realisations of the uncertainty sets [e.g., Ordóñez, 2010]. In practice, these assumptions usually result in optimisation models where the worst-case scenarios of the uncertain parameters (i.e., the limits of the uncertainty sets) must be considered in the problem, and the optimal solution must be feasible for all worst-case values that the uncertain parameters may take.

One of the main criticisms of robust optimisation is its excessive conservatism, which potentially leads to irrealistic and highly sub-optimal solutions. In order to control the degree of conservatism, the robust optimisation approach proposed by Bertsimas and Sim [2004] has been popular in the literature. It limits the minimum number of worst-case scenarios that a robust-feasible solution must be able to accommodate to a given value, Γ. The parameter Γ is typically called the budget of uncertainty. In these circumstances, a robust-feasible solution for a given Γ will be able to fathom up to Γ uncertain parameters taking their worst-case values, regardless of which parameters the worst-case values occur. To the best of our knowledge, robust optimisation has not yet been applied to the VRP with Synchronisation. This paper intends to bridge this gap by presenting a robust optimisation approach for this problem subject to uncertainty in vehicle travel times between customers. Given the temporal requirements present in synchronisation, it is comprehensible that vehicle travel times are a relevant parameter to be considered uncertain in this research work.

This paper leverages existing knowledge from the literature concerning robust optimisation and vehicle routing to obtain mathematical formulations for the Robust VRP with Synchronisation, which clarify and define this problem class. For solving larger instances, a branch-and-cut algorithm is implemented that considers the specificities of the problem being solved.

The remainder of this paper is as follows. Section 4.2 states and describes the Robust VRP with Synchronisation subject to uncertainty in travel times. Section 4.3 presents the solution method developed to solve this problem. Section 4.4 presents the obtained results of the computational experiments that were envisaged. Finally, Section 4.5 states the conclusions and provides insights into the developed work, as well as future research.

4.2. Problem description and formulations

This section outlines the different models for the Vehicle Routing Problem with Synchronisation, which will be utilised in this paper. Different model variants will be presented, which will include a general 3-index model and a simplified 2-index model.

Building on the modelling framework for the Vehicle Routing Problem with Synchronisation of Soares et al. [2022] and the robust optimisation approach of Munari et al. [2019] for the Vehicle Routing Problem with Time Windows, let us consider a problem whose purpose is to determine the optimal routes for a set of K vehicles, initially located at a depot 0, in order to perform a set of n tasks in a set of m locations, geographically dispersed.

Using as a basis the notation adopted by Munari et al. [2019], the travel times between tasks are subject to the *polyhedral uncertainty set* defined in Equation (4.1):

$$\mathscr{U} = \left\{ \tilde{d}' \in \mathbb{R}_+^{|\mathscr{E}|} : \tilde{d}'_{ij} = d'_{ij} + \hat{d}'_{ij}\xi_{ij}, \sum_{(i,j)\in\mathscr{E}} \xi_{ij} \leq \Gamma, 0 \leq \xi_{ij} \leq 1, \forall (i,j) \in \mathscr{E} \right\} \tag{4.1}$$

with d'_{ij} representing the nominal travel time between tasks i and j, $\tilde{d}'_{ij}$, representing the

random variables for the values of travel times between i and j and $\hat{d}'_{ij}$ representing the positive travel time deviations from their nominal values, d'_{ij}.
Let $k \in \mathcal{K}$ be a route that visits a sequence of $r+1$ tasks $(v_0, v_1, ..., v_r)$, where $v_0 = 0$ and $v_r = n+1$. In this context, let $t^{k\gamma}_{v_j}$ be the earliest exact time the service can start at node v_j when up to $\gamma \leq \Gamma$ travel times reach their worst-case values.
With this in mind, the Robust VRP with Synchronisation is defined through two alternative mathematical formulations: a general 3-index formulation and a simplified formulation with only two indices for the routing variables. Each one of these formulations will now be presented.

4.2.1 General three-index formulation for the robust problem

Sets

General sets:

$\mathcal{L}$	Set of locations
$\mathcal{K}$	Set of vehicles or routes, $\mathcal{K} = \{k_1, k_2, ..., k_K\}$
$\mathcal{N}$	Set of tasks, $\mathcal{N} = \{1, 2, ..., n\}$
$\mathcal{N}_0$	Set of tasks with depot start and end tasks, $\mathcal{N}_0 = \mathcal{N} \cup \{0, n+1\}$
$\mathcal{A}$	Set of arcs $(i, j) \in \mathcal{N}_0 \times \mathcal{N}_0$
$\mathcal{R}$	Set of all operations (i, j), $\mathcal{R} \subseteq \mathcal{N}_0 \times \mathcal{N}_0$

Parameters

General parameters:

c^k_{ij}	Cost for traversing arc $(i, j) \in \mathcal{A}$ by vehicle $k \in \mathcal{K}$
d_{ij}, d'_{ij}	Travel distance, time from location $\psi(i)$ to location $\psi(j)$
$\hat{d}_{ij}$	Maximum positive deviation of the travel time when travelling between tasks i and j
Q^k	Capacity of vehicle $k \in \mathcal{K}$
D^k	Maximum route length for route $k \in \mathcal{K}$
T	Duration of the problem's planning horizon

Parameters related with tasks:

a_i, b_i	Earliest, latest possible time to begin performing task $i \in \mathcal{N}_0$
e_i	takes value 1, is task i is mandatory; 0, otherwise
q_i	Demand to be satisfied for task $i \in \mathcal{N}_0$
r^k_i	takes value 1, if task $i \in \mathcal{N}_0$ can be performed by vehicle $k \in \mathcal{K}$; 0, otherwise
s_i	Service time of task $i \in \mathcal{N}$

Parameters related with operations:

e_{ij}	takes value 1, if operation $(i,j) \in \mathcal{R}$ is mandatory; 0, otherwise
v_{ij}	takes value 1, if operation (i,j) can be performed by the same vehicle; 0, otherwise
z_{ij}	takes value 1, if task j of operation (i,j) can be performed immediately after i; 0, otherwise

Parameters related with operation synchronisation:

w_{ij}	takes value 1, if operation (i,j) can be performed by the different vehicles; 0, otherwise
λ_{ij}	Time offset for lower-bounded sync. of operation $(i,j) \in \mathcal{R}$
μ_{ij}	Time offset for upper-bounded sync. of operation $(i,j) \in \mathcal{R}$

Transportation network

The transportation network is defined through $\mathcal{G} = (\mathcal{N}_0, \mathcal{A})$, with $\mathcal{A} \subseteq \mathcal{N}_0 \times \mathcal{N}_0$. The network can be simplified by applying several preprocessing procedures.

$$\mathcal{A} = \left\{ (i,j) \in \mathcal{N}_0 \times \mathcal{N}_0 : z_{ij} \sum_{k \in \mathcal{K}} r_i^k r_j^k \geq 1, a_i + s_i + d'_{ij} \leq b_j \right\} \tag{4.2}$$

Decision variables

We consider the following decision variables:

$$x_{ij}^k \quad \begin{cases} 1 & \text{if arc } (i,j) \in \mathcal{A} \text{ is traversed by vehicle } k \in \mathcal{K} \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

$$y_{ij}^{kk'} \quad \begin{cases} 1 & \text{if operation } (i,j) \in \mathcal{R} \text{ is performed by routes } k \text{ and } k', \text{ respectively} \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

$$u_{ij}^k \quad \text{Load of vehicle } k \in \mathcal{K} \text{ when traversing arc } (i,j) \in \mathcal{A} \tag{4.5}$$

$$t_i^{k\gamma} \quad \text{Earliest arrival time of vehicle } k \in \mathcal{K} \text{ at task } i \in \mathcal{N}_0 \text{ when } \gamma \leq \Gamma \text{ travel times have so far reached their worst-case values} \tag{4.6}$$

Note: Variables $y_{ij}^{kk'}$ may be relaxed to continuous variables without loss of model consistency, thus improving model tractability.

Objective function

The objective function of the problem consists in the minimisation of the total travelled distance (Eq. 4.7).

$$\min \sum_{k\in\mathcal{K}} \sum_{(i,j)\in\mathcal{A}} c_{ij}^k x_{ij}^k \tag{4.7}$$

Constraints

Fundamental routing constraints:

$$e_j \leq \sum_{k\in\mathcal{K}} \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k \leq 1 \qquad \forall j \in \mathcal{N} \tag{4.8}$$

$$\sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k - \sum_{i:(j,i)\in\mathcal{A}} x_{ji}^k = 0 \qquad \forall j \in \mathcal{N}, k \in \mathcal{K} \tag{4.9}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j}^k = \sum_{i:(i,n+1)\in\mathcal{A}} x_{i,n+1}^k \qquad \forall k \in \mathcal{K} \tag{4.10}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j}^k \leq 1 \qquad \forall k \in \mathcal{K} \tag{4.11}$$

Constraints (4.8) ensure that every task is performed at most once, and they force mandatory tasks to be performed; they are *global task constraints*. In this formulation, we assume that there are no tasks being performed more than once. Constraints (4.9) establish the inflow and outflow conservation: vehicles entering a task node must also exit it. Constraints (4.10) and (4.11) ensure that every vehicle starts and ends its route at the depot.

Constraints for earliest arrival times and time windows:

$$t_j^{k\gamma} \leq T \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k \qquad \forall j \in \mathcal{N}_0, k \in \mathcal{K}, \gamma = 0, ..., \Gamma \tag{4.12}$$

$$t_i^{k\gamma} + \left(s_i + d'_{ij}\right) x_{ij}^k \leq t_j^{k\gamma} + T(1 - x_{ij}^k) \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K}, \gamma = 0, ..., \Gamma \tag{4.13}$$

$$t_i^{k(\gamma-1)} + \left(s_i + d'_{ij} + \hat{d}'_{ij}\right) x_{ij}^k \leq t_j^{k\gamma} + T(1 - x_{ij}^k) \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K}, \gamma = 1, ..., \Gamma \tag{4.14}$$

$$\sum_{i:(i,j)\in\mathcal{A}} a_j x_{ij}^k \leq t_j^{k\gamma} \leq \sum_{i:(i,j)\in\mathcal{A}} b_j x_{ij}^k \qquad \forall j \in \mathcal{N}, k \in \mathcal{K}, \gamma = 0, ..., \Gamma \tag{4.15}$$

Constraints (4.12) are linking constraints between variables x_{ij}^k and $t_i^{k\gamma}$ at each value of γ; they impose that the earliest exact time the task can start cannot be different from zero if the vehicle does not perform said task $\left(x_{ij}^k = 0 \implies t_i^{k\gamma} = 0\right)$. Constraints (4.13)–(4.14) establish vehicles earliest arrival times to tasks. They also serve as sub-tour elimination constraints and are an alternative to the Miller-Tucker-Zemlin constraints. Constraints (4.15) establish lower and upper bounds to the earilest arrival time of a vehicle to a given task, according to its desired time windows.

Capacity constraints:

$$u_{ij}^k \leq Q^k x_{ij}^k \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K} \tag{4.16}$$

$$\sum_{i:(i,j)\in\mathcal{A}} u_{ij}^k + q_j \sum_{i:(i,j)\in\mathcal{A}} x_{ij}^k \le \sum_{i:(i,j)\in\mathcal{A}} u_{ji}^k \qquad \forall j \in \mathcal{N}_0 \setminus \{0\}, k \in \mathcal{K} \tag{4.17}$$

Constraints (4.16) are linking constraints between variables x_{ij}^k and u_{ij}^k; they impose that the load of vehicle k when traversing arc (i, j) cannot be different from zero if the vehicle does not traverse said arc $\left(x_{ij}^k = 0 \implies u_{ij}^k = 0\right)$. Constraints (4.17) establish the load of vehicle k when entering and leaving a task: the difference between the loads must account for the demand of the task being performed.

Constraints for setting decision variables $y_{ij}^{kk'}$:

$$\sum_{l:(l,i)\in\mathcal{A}} x_{li}^k + \sum_{l:(l,j)\in\mathcal{A}} x_{lj}^{k'} - 1 \le y_{ij}^{kk'} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{4.18}$$

$$y_{ij}^{kk'} \le \sum_{l:(l,i)\in\mathcal{A}} x_{li}^k \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{4.19}$$

$$y_{ij}^{kk'} \le \sum_{l:(l,j)\in\mathcal{A}} x_{lj}^{k'} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{4.20}$$

Constraints (4.18)-(4.20) are linking constraints that set the values of decision variables $y_{ij}^{kk'}$ based on the values of variables x_{ij}^k. Constraints (4.18) set variable $y_{ij}^{kk'}$ to 1 if tasks i and j of operation $(i, j) \in \mathcal{R}$ are being performed by its corresponding vehicles k and k'. Constraints (4.19) and (4.20) set the opposite case: when one of the tasks is not being performed, then the value of $y_{ij}^{kk'}$ is forcefully set to zero.

Global operation constraints:

$$e_{ij} \le \sum_{k\in\mathcal{K}} \sum_{k'\in\mathcal{K}} y_{ij}^{kk'} \le 1 \qquad \forall (i,j) \in \mathcal{R} \tag{4.21}$$

$$y_{ij}^{kk} = 0 \qquad \forall (i,j) \in \mathcal{R}, k \in \mathcal{K} : v_{ij} = 0 \tag{4.22}$$

$$y_{ij}^{kk'} = 0 \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} : k \neq k', w_{ij} = 0 \tag{4.23}$$

Constraints (4.21) ensure that operations be performed at most once, and they force mandatory operations to be performed, similarly to what happens in constraints (4.8). Constraints (4.22) are applied to operations that cannot be performed by the same vehicle: in these situations, if the operation is performed, then it must forcefully be performed by different vehicles. Constraints (4.23) are applied to operations that cannot be performed by different vehicles: in these situations, if the operation is performed, then it must forcefully be performed by the same vehicle.

Constraints for operation synchronisation:

$$t_i^{k\gamma} + \lambda_{ij} \le t_j^{k'\gamma'} + T\left(1 - y_{ij}^{kk'}\right) \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} : k \neq k', w_{ij} = 1, \gamma, \gamma' = 0, ..., \Gamma \tag{4.24}$$

$$t_i^{k\gamma} + \mu_{ij} + T\left(1 - y_{ij}^{kk'}\right) \geq t_j^{k'\gamma'} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} : k \neq k', w_{ij} = 1, \gamma, \gamma' = 0, ..., \Gamma \tag{4.25}$$

Constraints (4.24) ensure that, for a given synchronised operation (i, j), the start time of task j can only be performed λ_{ij} time units after the start time of task i. On the other hand, constraints (4.25) state that, for a given synchronised operation (i, j), if i is performed before j, then task j must start being performed up to μ_{ij} time units after i begins to be performed.

Decision variables domain:

$$x_{ij}^k \in \left\{0, r_i^k r_j^k\right\} \qquad \forall (i,j) \in \mathcal{A}, k \in \mathcal{K} \tag{4.26}$$

$$y_{ij}^{kk'} \in \left\{0, r_i^k r_j^{k'}\right\} \qquad \forall (i,j) \in \mathcal{R}, k, k' \in \mathcal{K} \tag{4.27}$$

$$0 \leq t_i^{k\gamma} \leq r_i^k T \qquad \forall i \in \mathcal{N}_0, k \in \mathcal{K}, \gamma = 0, ..., \Gamma \tag{4.28}$$

$$0 \leq u_{ij}^k \leq r_i^k r_j^k Q^k \qquad \forall i \in \mathcal{N}_0, k \in \mathcal{K} \tag{4.29}$$

Finally, constraints (4.26)–(4.29) establish the decision variables domain. The addition of parameter r_i^k to the bounds of the decision variables allows the model to only consider the vehicle-task combinations that are compatible by forcing variables to take value 0 when a given task i cannot be performed by vehicle k (i.e., $r_i^k = 0$).

4.2.2 Simplified, two-index formulation for the robust problem

This simplified mathematical formulation is only applicable to a VRP with Synchronisation under the following conditions:

- the only synchronisation type present in the entire problem consists of exact operation synchronisation;
- there is a homogeneous fleet, and therefore there are no compatibility constraints between tasks and vehicles;
- there is no enforcement of a maximum route length.

For the present formulation, most sets and parameters can be repurposed from the general 3-index formulation. Set $\mathcal{K}$ is no longer relevant and parameters D^k, r_i^k, v_{ij}, z_{ij}, δ_{ij} and Δ_{ij} are discarded. Parameters c_{ij}^k and Q^k are replaced by the following simplified parameters:

c_{ij}	Cost for traversing arc $(i, j) \in \mathcal{A}$
Q	Capacity of each vehicle

Transportation network

The transportation network is defined through $\mathcal{G} = (\mathcal{N}_0, \mathcal{A})$, with $\mathcal{A} \subseteq \mathcal{N}_0 \times \mathcal{N}_0$. The network can be simplified by applying several preprocessing procedures.

$$\mathcal{A} = \left\{(i,j) \in \mathcal{N}_0 \times \mathcal{N}_0 : z_{ij} = 1, a_i + s_i + d'_{ij} \le b_j\right\} \tag{4.30}$$

Decision variables

We consider the following decision variables:

$$x_{ij} \quad \begin{cases} 1 & \text{if arc } (i,j) \in \mathcal{A} \text{ is traversed} \\ 0 & \text{otherwise} \end{cases} \tag{4.31}$$

$$y_{ij} \quad \begin{cases} 1 & \text{if operation } (i,j) \in \mathcal{R} \text{ is performed} \\ 0 & \text{otherwise} \end{cases} \tag{4.32}$$

u_{ij} Load of vehicle when traversing arc $(i,j) \in \mathcal{A}$ (4.33)

t_i^γ Earliest arrival time at task $i \in \mathcal{N}_0$ when $\gamma \le \Gamma$ travel times have so far reached their worst-case values (4.34)

Note: Variables y_{ij} may be relaxed to continuous variables without loss of model consistency, thus improving model tractability.

Objective function

The objective function of the problem consists in the minimisation of the total travelled distance (Eq. 4.35).

$$\min \sum_{(i,j)\in\mathcal{A}} c_{ij} x_{ij} \tag{4.35}$$

Constraints

Fundamental routing constraints:

$$e_j \le \sum_{i:(i,j)\in\mathcal{A}} x_{ij} \le 1 \qquad \forall j \in \mathcal{N} \tag{4.36}$$

$$\sum_{i:(i,j)\in\mathcal{A}} x_{ij} - \sum_{i:(j,i)\in\mathcal{A}} x_{ji} = 0 \qquad \forall j \in \mathcal{N} \tag{4.37}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j} = \sum_{i:(i,n+1)\in\mathcal{A}} x_{i,n+1} \tag{4.38}$$

$$\sum_{j:(0,j)\in\mathcal{A}} x_{0j} \le K \tag{4.39}$$

Constraints (4.36) ensure that every task is performed at most once, and they force mandatory tasks to be performed; they are *global task contraints*. In this formulation, we assume that there are no tasks being performed more than once. Constraints (4.37) establish the inflow and outflow conservation: vehicles entering a task node must also exit it. Constraints (4.38) and (4.39) ensure that every vehicle starts and ends its route at the depot.

Capacity constraints:

$$u_{ij} \leq Qx_{ij} \qquad \forall (i,j) \in \mathcal{A} \tag{4.40}$$

$$\sum_{i:(i,j)\in\mathcal{A}} u_{ij} + q_j \sum_{i:(i,j)\in\mathcal{A}} x_{ij} \leq \sum_{i:(i,j)\in\mathcal{A}} u_{ji} \qquad \forall j \in \mathcal{N}_0 \setminus \{0\} \tag{4.41}$$

Constraints (4.40) are linking constraints between variables x_{ij} and u_{ij}; they impose that the load of a vehicle when traversing arc (i, j) cannot be different from zero if no vehicle traverses said arc $\left(x_{ij} = 0 \implies u_{ij} = 0\right)$. Constraints (4.41) establish the load of a vehicle when entering and leaving a task: the difference between the loads must account for the demand of the task being performed.

Constraints for task start times and time windows:

$$t_j^\gamma \leq T \sum_{i:(i,j)\in\mathcal{A}} x_{ij} \qquad \forall j \in \mathcal{N}_0, \gamma = 0, ..., \Gamma \tag{4.42}$$

$$t_i^\gamma + \left(s_i + d'_{ij}\right) x_{ij} \leq t_j^\gamma + T(1 - x_{ij}) \qquad \forall (i,j) \in \mathcal{A}, \gamma = 0, ..., \Gamma \tag{4.43}$$

$$t_i^{\gamma-1} + \left(s_i + d'_{ij} + \hat{d}'_{ij}\right) x_{ij} \leq t_j^\gamma + T(1 - x_{ij}) \qquad \forall (i,j) \in \mathcal{A}, \gamma = 1, ..., \Gamma \tag{4.44}$$

$$\sum_{i:(i,j)\in\mathcal{A}} a_j x_{ij} \leq t_j^\gamma \leq \sum_{i:(i,j)\in\mathcal{A}} b_j x_{ij} \qquad \forall j \in \mathcal{N}, \gamma = 0, ..., \Gamma \tag{4.45}$$

Constraints (4.42) are linking constraints between variables x_{ij} and t_i^γ at each value of γ; they impose that the earliest arrival time of vehicle k at task i cannot be different from zero if the vehicle does not perform said task $\left(x_{ij} = 0 \implies t_i^\gamma = 0\right)$. Constraints (4.43)–(4.44) establish vehicles earliest arrival times to task nodes. They also serve as sub-tour elimination constraints and are an alternative to the Miller-Tucker-Zemlin constraints. Constraints (4.45) establish lower and upper bounds to the earliest arrival time of a vehicle to a given task, according to its desired time windows.

Constraints for setting decision variables y_{ij}:

$$\sum_{l:(l,i)\in\mathcal{A}} x_{li} + \sum_{l:(l,j)\in\mathcal{A}} x_{lj} - 1 \leq y_{ij} \qquad \forall (i,j) \in \mathcal{R} \tag{4.46}$$

$$y_{ij} \leq \sum_{l:(l,i)\in\mathcal{A}} x_{li} \qquad \forall (i,j) \in \mathcal{R} \tag{4.47}$$

$$y_{ij} \leq \sum_{l:(l,j)\in\mathcal{A}} x_{lj} \qquad \forall (i,j) \in \mathcal{R} \tag{4.48}$$

Constraints (4.46)-(4.48) are linking constraints that set the values of decision variables y_{ij} based on the values of variables x_{ij}. Constraints (4.46) set variable y_{ij} to 1 if tasks i and j

of operation $(i, j) \in \mathcal{R}$ are being performed. Constraints (4.47) and (4.48) set the opposite case: when one of the tasks is not being performed, then the value of y_{ij} is forcefully set to zero.

Global operation constraints:

$$e_{ij} \leq y_{ij} \leq 1 \qquad \forall (i, j) \in \mathcal{R} \tag{4.49}$$

Constraints (4.49) ensure that operations be performed at most once, and they force mandatory operations to be performed, similarly to what happens in constraints (4.36).

Constraints for operation synchronisation:

$$t_i^{\gamma} + \lambda_{ij} \leq t_j^{\gamma'} + T\left(1 - y_{ij}\right) \qquad \forall (i, j) \in \mathcal{R}, \gamma, \gamma' = 0, ..., \Gamma \tag{4.50}$$

$$t_i^{\gamma} + \mu_{ij} + T\left(1 - y_{ij}\right) \geq t_j^{\gamma'} \qquad \forall (i, j) \in \mathcal{R}, \gamma, \gamma' = 0, ..., \Gamma \tag{4.51}$$

Constraints (4.50) ensure that, for a given synchronised operation (i, j), the earliest start time of task j can only be performed λ_{ij} time units after the start time of task i. On the other hand, constraints (4.51) state that, for a given synchronised operation (i, j), if i is performed before j, then the earliest start time of task j can only be performed μ_{ij} time units before the start time of i.

Decision variables domain:

$$x_{ij} \in \{0, 1\} \qquad \forall (i, j) \in \mathcal{A} \tag{4.52}$$

$$y_{ij} \in \{0, 1\} \qquad \forall (i, j) \in \mathcal{R} \tag{4.53}$$

$$0 \leq t_i^{\gamma} \leq T \qquad \forall i \in \mathcal{N}_0, \gamma = 0, ..., \Gamma \tag{4.54}$$

$$0 \leq u_{ij} \leq Q \qquad \forall i \in \mathcal{N}_0 \tag{4.55}$$

Finally, constraints (4.52)–(4.55) establish the decision variables domain.

4.3. Branch-and-cut algorithm

Due to the Robust VRP with Synchronisation being still a new problem, it was deemed appropriate to develop an exact approach. This is due to the fact that exact approaches allow us to obtain more information regarding the quality of the solution process and of the obtained solutions, namely their distance from optimality. Therefore, the mathematical formulation presented earlier served as a basis for developing a branch-and-cut algorithm. In general traces, branch-and-cut approaches leverage the branch-and-bound algorithm with the progressive addition of cutting planes as the solution process evolves. The algorithm initially starts with a simplified model where given sets of constraints are omitted. The branch-and-bound procedure is implemented by resorting to a commercial solver.
As the branch-and-bound procedure embedded in the solver finds feasible solutions for

the simplified model, separation algorithms are invoked to check the solution for possible violations of constraints that were omitted, in which case additional constraints will be introduced into the model and the updated model is solved again. The constraints being introduced must be structured in a way that necessarily prohibits the solver from accepting the infeasible solution. This process is repeated until no separation algorithms can detect violated constraints, in which case we successfully obtain a feasible solution for the problem.

4.3.1 Initialisation

For the problem at hand, the initial model that is fed to the commercial solver consists of the model presented in section 4.2.2 omitting the following constraints:

- Capacity constraints (4.40)–(4.41);
- Constraints for task start times and time windows (4.42)–(4.45);
- Constraints for operation synchronisation (4.50)–(4.51).

In practice, the removal of these constraints turns the initial model into an uncapacitated VRP. However, several initialisation procedures are still invoked before algorithm startup. Furthermore, a set of initial constraints are also introduced into the initial model in order to facilitate convergence.

Arc elimination. This procedure relies on a time window tightening procedure, where parameters a_i and b_i, which translate the lower and upper bounds of the time window of task i, respectively, are repurposed to calculate differentiated time window bounds dependent on the arcs that are traversed.
With this in mind, the earliest and latest arrival times a_i and b_i are used to generate parameters a_{ij} and b_{ij}. Parameters a_{ij} translate the earliest arrival time at task j when $(i, j) \in \mathcal{A}$ is traversed. Analogously, parameters b_{ij} represent the latest arrival time at task j when $(j,i) \in \mathcal{A}$ is traversed.
They are calculated as follows:

$$a_{ij} = \max\left\{a_j, a_i + s_i + d'_{ij}\right\} \qquad \forall (i, j) \in \mathcal{A}$$
$$b_{ij} = \min\left\{b_j, b_i - d'_{ji} - s_j\right\} \qquad \forall (j,i) \in \mathcal{A}$$

This procedure was also expanded to consider the robust optimisation aspects of the problem. Parameter $\hat{a}_{ij}$ translates the earliest arrival time at task j when $(i, j) \in \mathcal{A}$ is traversed and $\tilde{d}'_{ij}$ takes its worst-case value. Analogously, parameter $\hat{b}_{ij}$ represents the latest arrival time at task j when $(j,i) \in \mathcal{A}$ is traversed and $\tilde{d}'_{ji}$ takes its worst-case value.

$$\hat{a}_{ij} = \max\left\{a_j, a_i + s_i + d'_{ij} + \hat{d}'_{ij}\right\} \qquad \forall (i, j) \in \mathcal{A}$$
$$\hat{b}_{ij} = \min\left\{b_j, b_i - \hat{d}'_{ji} - d'_{ji} - s_j\right\} \qquad \forall (j,i) \in \mathcal{A}$$

Taking into account these parameters, the set of arcs is preprocessed by subjecting it to an arc elimination procedure that evaluates the following rules for each arc $(i, j) \in \mathcal{A}$:

$$
\begin{aligned}
a_{ij} > b_j \quad &\Longrightarrow \quad \text{arc } (i,j) \text{ is infeasible} \\
\hat{a}_{ij} > b_j \wedge \Gamma > 0 \quad &\Longrightarrow \quad \text{arc } (i,j) \text{ is infeasible} \\
b_{ji} < a_i \quad &\Longrightarrow \quad \text{arc } (i,j) \text{ is infeasible} \\
\hat{b}_{ji} > a_j \wedge \Gamma > 0 \quad &\Longrightarrow \quad \text{arc } (i,j) \text{ is infeasible}
\end{aligned}
$$

After evaluating all of the arcs of the transportation network, the ones that were deemed infeasible are removed from the problem.

Infeasible paths. After performing the arc elimination procedure, the entire routing network is combed through an enumeration procedure of consecutive arcs of set $\mathcal{A}$ to find small-size infeasible paths.
This infeasible path identification procedure takes into account the robust optimisation aspects of the problem concerning the travel times between tasks. Iterating through each path $\mathcal{R} = (r_1, ..., r_{|R|})$ of length $|R|$, the hypothetical arrival times are computed for each task in this path and checked for its feasibility within the established time window bounds. If at least one infeasibility is found, the tested path is infeasible. For paths of length $|R| = 3$, this procedure can be simplified by iterating through each path (i, j, l), with $(i, j), (j, l) \in \mathcal{A}$, and applying the following rules of thumb instead of having to compute the arrival times:

$$
\begin{aligned}
\Gamma \geq 2 \wedge \hat{a}_{ij} > \hat{b}_{lj} \quad &\Longrightarrow \quad \text{path } (i,j,l) \text{ is infeasible} \\
\Gamma = 1 \wedge (\hat{a}_{ij} > b_{lj} \vee a_{ij} > \hat{b}_{lj}) \quad &\Longrightarrow \quad \text{path } (i,j,l) \text{ is infeasible} \\
\Gamma = 0 \wedge a_{ij} > b_{jl} \quad &\Longrightarrow \quad \text{path } (i,j,l) \text{ is infeasible}
\end{aligned}
$$

The infeasible paths that result from this procedure are used for introducing valid inequalities, described ahead in this paper.

Reversible infeasible paths. We define an infeasible path $R = (r_1, ..., r_{|R|})$ as reversible if it is also infeasible for all its possible path permutations of length $|R|$.
If we are able to find reversible infeasible paths in the procedure described above, we flag these paths as reversible so that they can be subject to tighter valid inequalities, described ahead in this paper.

4.3.2 Valid inequalities

After the initialisation steps described earlier, the initial model is added a limited number of constraints, some of which rely on the time-window tightening procedure that was previously described.

Cutting small deterministic and robust-infeasible paths. Taking into account the list of infeasible paths that were previously determined in the enumeration procedure described earlier, a valid inequality is introduced for each one of these paths. Considering a path $R = (r_1, ..., r_{|R|})$, of length R, that is deemed infeasible, the following valid inequalities (4.56) are valid:

$$\sum_{i=1}^{|R|-1} x_{r_i,r_{i+1}} \leq |R| - 2 \tag{4.56}$$

If, for path R, it is also possible to infer that it is a reversible infeasible path, then tighter constraints (4.57) can be entered instead of valid inequalities (4.56).

$$\sum_{i,j \in R} x_{ij} \leq |R| - 2 \tag{4.57}$$

Strengthened subtour elimination constraints based on synchronisation constraints. Taking into account the operations that are subject to synchronisation, it is possible to derive strengthened subtour elimination constraints. Let us consider set $\mathcal{S}$, which is defined as the powerset of $\mathcal{N}$, the set of all tasks, and S an element of $\mathcal{S}$. Let us also consider set $\mathcal{R}' = \{(a,b) \in \mathcal{R} : e_{ab} = 1 \wedge v_{ab} = 0 \wedge w_{ab} = 1\}$, which contains all mandatory operations that are subject to synchronisation.
If $\exists (a,b) \in \mathcal{R}' : a \in S \wedge b \in S$, the following constraint (4.58) can be introduced:

$$\sum_{i,j \in S} x_{ij} \leq |S| - 2 \tag{4.58}$$

If $\nexists (a,b) \in \mathcal{R}' : a \in S \wedge b \in S$ and, instead, $\exists (a,b) \in \mathcal{R}' : a \notin S \wedge b \in S$, the following constraint (4.59) can be introduced:

$$\sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{aj} + \sum_{i \in S} x_{ia} \leq |S| - 1 \tag{4.59}$$

If $\nexists (a,b) \in \mathcal{R}' : ((a \in S \wedge b \in S) \vee (a \notin S \wedge b \in S))$ and, instead, $\exists (a,b) \in \mathcal{R}' : a \in S \wedge b \notin S$, the following constraint (4.60) can be introduced:

$$\sum_{i,j \in S} x_{ij} + \sum_{j \in S} x_{bj} + \sum_{i \in S} x_{ib} \leq |S| - 1 \tag{4.60}$$

If, instead, $\nexists (a,b) \in \mathcal{R}' : (a \in S \wedge b \in S) \vee (a \notin S \wedge b \in S) \vee (a \in S \wedge b \notin S)$, the general case of subtour elimination constraints applies, presented in constraint (4.61):

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \tag{4.61}$$

Additional subtour elimination constraints can be introduced in parallel with the ones above. Let us consider set $\mathcal{S}$, which is defined as being the powerset of $\mathcal{N}$, the set of all tasks, and S an element of $\mathcal{S}$.
Let us also consider an auxiliary set $\mathcal{T} = \{\{a,b\} : (a,b) \in \mathcal{R}' \wedge a \in S \wedge b \in S\}$, which contains all the sets of synchronised tasks that are present in S. Furthermore, $T = \{a : \{a,b\} \in \mathcal{T}, \forall b\}$ is defined by containing all tasks in S that are subject to synchronisation inside S itself. Additionally, let $\mathcal{U}_i = \{j \in S : (i,j) \in \mathcal{A}\}$ be auxiliary sets which translate the tasks within S that are directly accessible from $i \in S$. If $|T| = |S|$, i.e., set S only contains synchronised operations within S, then the constraints (4.62)–(4.63) can be introduced:

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq |T| - \max_{i \in S} \{|\mathcal{U}_i|\} \tag{4.62}$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq |T| - \max_{i \in S} \{|\mathcal{U}_i|\} \tag{4.63}$$

Strengthened capacity constraints based on synchronisation requirements. These valid inequalities intend to cut infeasible solutions taking into account task capacity and synchronisation constraints. Let us consider set $\mathcal{S}$, which is defined as being the powerset of $\mathcal{N}$, the set of all tasks, and S an element of $\mathcal{S}$.
Let us also consider an auxiliary set $\mathcal{T} = \{\{a,b\} : (a,b) \in \mathcal{R} \wedge a \in S \wedge b \in S\}$, which contains all the sets of synchronised tasks that are present in S. Furthermore, $T = \{a : \{a,b\} \in \mathcal{T}, \forall b\}$ is defined by containing all tasks in S that are subject to synchronisation inside S itself.
Additionally, let $q(S) = \sum_{i \in S} q_i$ and $q(T) = \sum_{i \in T} q_i$ be the total load associated to sets S and T, respectively. Also, let $q'(\mathcal{T}) = \sum_{\{a,b\} \in \mathcal{T}} (\min\{q_a, q_b\})$ be the sum of the lightest load in each synchronised operation.
If $T \neq \emptyset$ and it is possible to ensure that $|T| = 2|\mathcal{T}|$ (i.e., there are no "crossed" synchronisation constraints), then the following constraints (4.64)–(4.65) can be introduced:

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq \max\left(2\left\lceil \frac{q'(\mathcal{T})}{Q} \right\rceil, \left\lceil \frac{q(S)}{Q} \right\rceil\right) \tag{4.64}$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq \max\left(2\left\lceil \frac{q'(\mathcal{T})}{Q} \right\rceil, \left\lceil \frac{q(S)}{Q} \right\rceil\right) \tag{4.65}$$

with Q being the vehicle capacity.
Instead, if $T = \emptyset$ or $|T| < 2|\mathcal{T}|$, the general case of rounded capacity constraints applies, as in constraints (4.66)–(4.67):

$$\sum_{i \notin S} \sum_{j \in S} x_{ij} \geq \left\lceil \frac{q(S)}{Q} \right\rceil \tag{4.66}$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq \left\lceil \frac{q(S)}{Q} \right\rceil \tag{4.67}$$

4.3.3 Separation routines

Whenever the solver can find a candidate solution, its feasibility must be checked against all requirements that are not still embedded into the model. To that effect, a list of separation routines is executed to identify such infeasibilities so that cutting planes can afterwards be derived and added to the model. For each one of the candidate solutions found by the solver, the separation routines are executed sequentially in the order in which they are described in this paper. If a routine is able to find an infeasibility, the algorithm skips subsequent separation routines and proceeds to construct the cuts associated with the newly-found infeasibility.
Each one of the separation routines that were implemented is now described.

4.3.3.1 Identification of subtours

Given a candidate solution found by the solver, and since not all subtour elimination constraints are included in the initial model, there may occur some subtours within the solution that must be discarded for the solution to become feasible. Therefore, the first separation routine verifies if any subtours exist in the candidate solution, in which case an infeasibility is found and subtour elimination constraints will necessarily be added to the model.

4.3.3.2 Vehicle capacity infeasibilities

A separation routine is invoked to check if route capacity constraints are respected.
If it is verified that the total transported load by a given route is greater than the vehicle capacity, the separation routine identifies all (minimal) route sub-paths whose cumulative transported load exceeds the vehicle capacity, as exemplified in Table 4.1.

Table 4.1: Illustration of the infeasible path separation routine based on capacity constraints (vehicle capacity = 200.0)

Route	0	→ 5	→ 3	→ 12	→ 18	→ 1	→ 4	→ 20	→ 25	→ 17	→ 0
Task Demand	0.0	40.0	10.0	25.0	35.0	25.0	15.0	40.0	30.0	50.0	20.0
Infeasible path #1		5	→ 3	→ 12	→ 18	→ 1	→ 4	→ 20	→ **25**		
Cumulative Load		40.0	50.0	75.0	110.0	135.0	150.0	190.0	**220.0**		
Infeasible path #2			3	→ 12	→ 18	→ 1	→ 4	→ 20	→ 25	→ **17**	
Cumulative Load			10.0	35.0	70.0	95.0	110.0	150.0	180.0	**230.0**	

4.3.3.3 Infeasible paths based on operation constraints

Given a candidate solution found by the solver, the solution must be checked for operations (pairs of tasks) that should be performed by different routes but are instead being performed by the same route. If this situation occurs for a given operation (i, j), there

exists an infeasible path, $(i, ..., j)$ or $(j, ..., i)$, which must be added to the list of reversible infeasible paths.
This separation routine will ultimately generate the cuts that end up not being generated through the enumeration procedure described earlier.

4.3.3.4 Infeasible paths based on time window bounds

Given a candidate found by the solver, and after retrieving it, a data structure, visually exemplified in Table 4.2 considering $\Gamma = 5$, is populated by calculating the arrival times at tasks for each route. In practice, the data structure contains the values of variables t_i^γ, which were omitted from the base model.

Table 4.2: Infeasible path separation routine based on time window bounds ($\Gamma = 5$)

		Route																			
	0	→	2	→	8	→	10	→	11	→	9	→	6	→	4	→	1	→	18	→	0
a_i	0.0		20.0		255.0		357.0		410.0		155.0		550.0		727.0		605.0		522.0		0.0
b_i	950.0		412.0		630.0		876.0		634.0		544.0		650.0		765.0		836.0		950.0		950.0
Arrival Times, t_i^γ																					
γ = 0	0.0		20.6		255.0		357.0		450.0		543.1		639.0		727.0		820.5		929.1		979.1
1	0.0		30.9		255.0		357.0		451.5		544.6		641.7		727.0		822.0		938.4		988.4
2	0.0		30.9		255.0		357.0		451.5		546.1		641.7		727.0		823.5		939.9		989.9
3	0.0		30.9		255.0		368.0		451.5		546.1		641.7		727.0		825.0		941.4		991.4
4	0.0		30.9		255.0		368.0		451.5		546.1		641.7		727.0		826.1		942.9		992.9
5	0.0		30.9		255.0		368.0		451.5		546.1		641.7		727.0		827.2		944.0		994.0
$t_i^\Gamma = a_i$	1		0		1		0		0		0		0		1		0		0		0
$t_i^\Gamma \leq b_i$	1		1		1		1		1		0		0		1		1		1		0

infeasible path #1	8	→	10	→	1	→	9
infeasible path #2	4	→	1	→	18	→	0

In conjunction with this matrix of arrival times, two auxiliary vectors are also calculated and shown in Table 4.2. The first vector takes value 1 if the arrival time of task i at Γ, t_i^Γ, is equal to the TW lower bound, a_i. In these situations, it means that the arrival time of said task i did not depend on its preceding tasks; i.e., the preceding tasks had no impact on changing/increasing the arrival time of i. If $t_i^\Gamma > a_i$, the vector takes value 0.
The second vector takes value 1 if the arrival time of task i at Γ, t_i^Γ, is lower or equal to the TW upper bound, b_i; 0, otherwise. In practice, this vector translates if the arrival times respect TW constraints and therefore represents whether the route is feasible or not.
The infeasible path separation procedure occurs as follows.
Starting initially on 0:

1. We find the next task where $t_i^\Gamma > b_i$. This task will correspond to the end of the infeasible path.

2. Starting from the end of the infeasible path backwards, we find the first task whose arrival time $t_i^\Gamma = a_i$. This task will correspond to the start of the infeasible path.

3. If we've reached the end of the route, we stop. If not, we try to derive a new infeasible path by returning to step 1, starting from the next task after the end of the newly-found path whose arrival time $t_i^\Gamma = a_i$.

Using the example provided in Table 4.2, the first task where $t_i^\Gamma > b_i$ is task 9. Starting from task 9 backwards, the first task where $t_i^\Gamma = a_i$ is task 8. Therefore, the first infeasible path we are able to derive is 8–10–11–9. Starting from task 9, the next task where $t_i^\Gamma = a_i$ is task 4. Starting from this task, the next task where $t_i^\Gamma > b_i$ is task 0. Starting from this task backwards, the first task where $t_i^\Gamma = a_i$ is task 4. Therefore, the second infeasible path we are able to derive is 4–1–18–0.

4.3.3.5 Minimization of infeasible paths based on time window bounds

With infeasible paths determined, we may be able to shorten them further by removing some of its initial tasks. To that effect, we progressively shorten the path and apply the same heuristic described above. Table 4.3 exemplifies the shortening process of infeasible path #1. Considering sub-path 10–11–9, we set the arrival times of task 10 to the TW lower bound and calculate the subsequent arrival times. If we verify that t_9^5 remains greater than b_9, then path 8–10–11–9 can be shortened to 10–11–9. By trying to shorten the path even further, we conclude that we are unable to obtain an infeasible path, because for path 11–9 the arrival time at task 9 already respects time window bounds. Therefore, 10–11–9 is a minimal infeasible path.

4.3.3.6 Infeasible paths based on a single synchronisation constraint

Separation procedures were also developed to find infeasibilities in the arrival times matrix when a single synchronisation constraint is introduced.
Using one synchronisation constraint at a time, the arrival times matrix is updated to account for that single synchronisation constraint, using the following rules of thumb.

Lower-bounded synchronisation (LB). Considering an operation $(i, j) \in \mathcal{R}$ subject to lower-bounded synchronisation with a time offset λ_{ij}:

$$\text{if } t_i^{\gamma_1} + \lambda_{ij} > t_j^{\gamma_2} \qquad \Longrightarrow \qquad t_j^{\gamma_2} := \max\left\{t_j^{\gamma_2}, t_i^{\gamma_1} + \lambda_{ij}\right\} \qquad \forall \gamma_1 = 0, ..., \Gamma, \gamma_2 = 0, ..., \Gamma$$

Upper-bounded synchronisation (UB). Considering an operation $(i, j) \in \mathcal{R}$ subject to upper-bounded synchronisation with a time offset μ_{ij}:

$$\text{if } t_i^{\gamma_1} + \mu_{ij} < t_j^{\gamma_2} \qquad \Longrightarrow \qquad t_i^{\gamma_1} := \max\left\{t_i^{\gamma_1}, t_j^{\gamma_2} - \mu_{ij}\right\} \qquad \forall \gamma_1 = 0, ..., \Gamma, \gamma_2 = 0, ..., \Gamma$$

Table 4.3: Minimisation of previously identified infeasible paths

	Path		
	10 →	11 →	9
a_i	357.0	410.0	155.0
b_i	876.0	634.0	544.0
Arrival Times, t_i^γ			
γ = 0	357.0	450.0	543.1
γ = 1	357.0	455.5	544.6
γ = 2	357.0	455.5	546.1
γ = 3	357.0	455.5	546.1
γ = 4	357.0	455.5	546.1
γ = 5	357.0	455.5	546.1
$t_i^\Gamma = a_i$	1	0	0
$t_i^\Gamma \leq b_i$	1	1	0
inf. path # 1	10 →	11 →	9

	Path	
	11 →	9
a_i	410.0	155.0
b_i	634.0	544.0
Arrival Times, t_i^γ		
γ = 0	410.0	490.5
γ = 1	410.0	499.0
γ = 2	410.0	504.0
γ = 3	410.0	510.0
γ = 4	410.0	510.0
γ = 5	410.0	510.0
$t_i^\Gamma = a_i$	1	0
$t_i^\Gamma \leq b_i$	1	1
feasible path	11 →	9

As an example, let us consider two routes, 1 and 2, whose arrival times are represented in Table 4.4. In this particular instance, a lower-bounded (LB) synchronisation constraint must be applied between tasks 7 and 8, with $\lambda_{7,8} = 50.0$.

We conclude that the synchronisation constraint is not being respected with these arrival times. By applying the rules of thumb that were previously described, we will need to recalculate the arrival times of route 2, starting from task 8. This update procedure resulted in the arrival times that are represented in Table 4.5.

The update procedure rendered an infeasibility in route 2, meaning that we can conclude that this current solution cannot enforce this synchronisation constraint. Therefore, we are able to retrieve a pair of infeasible paths that cannot occur simultaneously in the same solution.

For route 1, the one that was not updated, the infeasible path will extend from task 2 (the last task before task 7 where $t_i^\Gamma = a_i$) to task 7 (the synchronised task itself). For route 2, the one whose arrival times were updated, the infeasible path will extend from task 6 (the last task before task 8 where $t_i^\Gamma = a_i$) to task 25 (the task whose arrival time was rendered infeasible by the synchronisation constraint).

Table 4.4: Illustration of the infeasible path separation routine based on synchronisation constraints

Route 1

		0	→ 2	→ 4	→ 10	→ 11	→ 7*	→ 15	→ 20	→ 21	→ 3	→ 0
a_i		0.0	75.0	255.0	357.0	410.0	532.0	422.0	727.0	635.0	533.0	0.0
b_i		995.0	412.0	630.0	876.0	634.0	600.0	733.0	831.0	855.0	995.0	995.0
Earliest arrival Times, t_i^γ												
γ	0	0.0	75.0	243.5	357.0	411.0	543.1	639.0	727.0	820.5	929.1	979.1
	1	0.0	75.0	243.5	357.0	416.5	544.6	641.7	727.0	822.0	938.4	988.4
	2	0.0	75.0	243.5	363.6	420.0	546.1	646.7	727.0	823.5	939.9	989.9
	3	0.0	75.0	265.0	368.0	431.5	546.1	646.7	727.0	825.0	941.4	991.4
	4	0.0	75.0	265.0	368.0	434.5	546.1	646.7	727.0	826.1	942.9	992.9
	5	0.0	75.0	265.0	373.5	439.0	546.1	646.7	727.0	827.2	944.0	994.0
$t_i^\Gamma = a_i$		1	1	0	0	0	0	0	1	0	0	0
$t_i^\Gamma \leq b_i$		1	1	1	1	1	1	1	1	1	1	1

Route 2

		0	→ 24	→ 6	→ 12	→ 13	→ 8*	→ 18	→ 22	→ 25	→ 0
a_i		0.0	15.0	255.0	357.0	422.0	404.0	510.0	627.0	655.0	522.0
b_i		995.0	557.0	388.0	498.0	466.0	635.0	770.0	788.0	770.0	995.0
Earliest arrival Times, t_i^γ											
γ	0	0.0	20.6	255.0	357.0	450.0	413.0	539.0	627.0	730.5	829.0
	1	0.0	30.9	255.0	357.0	451.5	424.0	541.7	627.0	742.0	838.0
	2	0.0	30.9	255.0	357.0	451.5	424.0	546.7	627.0	743.5	839.0
	3	0.0	30.9	255.0	368.0	451.5	427.5	546.7	627.0	755.0	841.0
	4	0.0	30.9	255.0	368.0	451.5	430.0	546.7	627.0	766.1	842.0
	5	0.0	30.9	255.0	368.0	451.5	441.5	546.7	627.0	767.2	844.0
$t_i^\Gamma = a_i$		1	0	1	0	0	0	0	1	0	0
$t_i^\Gamma \leq b_i$		1	1	1	1	1	1	1	1	1	1

Table 4.5: Illustration of the infeasible path separation routine based on synchronisation constraints after arrival times are updated

Route 1

	0 →	2 →	4 →	10 →	11 →	7* →	15 →	20 →	21 →	3 →	0
a_i	0.0	75.0	243.5	357.0	410.0	532.0	422.0	727.0	635.0	533.0	0.0
b_i	995.0	412.0	703.0	876.0	634.0	600.0	733.0	831.0	855.0	995.0	995.0
Earliest arrival Times, t_i^γ											
γ = 0	0.0	75.0	243.5	357.0	411.0	543.1	639.0	727.0	820.5	929.1	979.1
1	0.0	75.0	243.5	357.0	416.5	544.6	641.7	727.0	822.0	938.4	988.4
2	0.0	75.0	243.5	363.6	420.0	546.1	646.7	727.0	823.5	939.9	989.9
3	0.0	75.0	265.0	368.0	431.5	546.1	646.7	727.0	825.0	941.4	991.4
4	0.0	75.0	265.0	368.0	434.5	546.1	646.7	727.0	826.1	942.9	992.9
5	0.0	75.0	265.0	373.5	439.0	546.1	646.7	727.0	827.2	944.0	994.0
$t_i^\Gamma = a_i$	1	1	0	0	0	0	0	1	0	0	0
$t_i^\Gamma \leq b_i$	1	1	1	1	1	1	1	1	1	1	1

infeasible path: 2 → 4 → 10 → 11 → 7

Route 2

	0 →	24 →	6 →	12 →	13 →	8* →	18 →	22 →	25 →	0
a_i	0.0	15.0	255.0	357.0	422.0	404.0	510.0	627.0	655.0	522.0
b_i	995.0	557.0	388.0	498.0	466.0	635.0	770.0	788.0	770.0	995.0
Earliest arrival Times, t_i^γ										
γ = 0	0.0	20.6	255.0	357.0	450.0	***596.1***	***678.0***	***703.0***	***780.0***	***829.0***
1	0.0	30.9	255.0	357.0	451.5	***596.1***	***687.5***	***715.3***	***783.0***	***838.0***
2	0.0	30.9	255.0	357.0	451.5	***596.1***	***687.5***	***715.3***	***783.0***	***839.0***
3	0.0	30.9	255.0	368.0	451.5	***596.1***	***687.5***	***715.3***	***799.0***	***841.0***
4	0.0	30.9	255.0	368.0	451.5	***596.1***	***687.5***	***718.0***	***799.0***	***842.0***
5	0.0	30.9	255.0	368.0	451.5	***596.1***	***687.5***	***722.3***	***799.0***	***844.0***
$t_i^\Gamma = a_i$	1	0	1	0	0	0	0	0	0	0
$t_i^\Gamma \leq b_i$	1	1	1	1	1	1	1	1	0	1

infeasible path: 6 → 12 → 13 → 8 → 18 → 22 → 25

4.3.3.7 Infeasible paths based on multiple synchronisation constraints

The previous path separation routine considers only one synchronisation constraint at a time. However, there are some solutions whose infeasibility can only be verified when multiple synchronisation constraints are at play.
This separation procedure iteratively attempts to update the route arrival times after enforcing multiple synchronisation constraints. Table 4.6 shows a list of synchronisation constraints to be respected for a given problem instance.

Table 4.6: Illustration of the infeasible path separation routine based on multiple synchronisation constraints

#	Type	i	j	λ_{ij}	μ_{ij}	Task to update arrival times			Reference Task			Up to Date
						Task	Position	Route	Task	Position	Route	
7	LB	35	11	0.0	--	11	1	1	35	1	2	0
4	LB	45	31	0.0	--	31	1	3	45	5	4	0
5	LB	33	22	0.0	--	22	2	2	33	3	3	0
1	LB	15	42	0.0	--	42	2	4	25	5	1	0
6	UB	43	52	--	0.0	43	3	4	52	2	1	0
8	UB	14	27	--	0.0	14	4	1	27	4	2	0
2	UB	15	42	--	0.0	25	5	1	42	2	4	0
3	UB	45	31	--	0.0	45	5	4	31	1	3	0

Given that the synchronisation constraints that end up being selected in this separation routine depend on the concrete order of this list, the list can be shuffled a fixed number of times, in order to maximise the number of different infeasible combinations of synchronisation constraints.
The separation routine is executed as follows. Following the order of the previous table, each synchronisation constraint is checked for the candidate solution by updating the arrival times matrix as described in the previous separation routine. After updating, column "Up to Date" is changed to 1. If an infeasibility is found, the separation routine stops, and a new set of infeasible paths has been found. If not, the separation routine progresses to the next synchronisation constraint.
It should be noted that it may be necessary to iterate multiple times through the constraints list, since updating certain synchronisation constraints may change arrival times where other synchronisation constraints are involved.
Table 4.7 exemplifies this situation. In updating synchronisation constraint #1, the arrival times of route #4, starting in position #2, are recalculated, which means that synchronisation constraint #4 must be re-updated, because one of its tasks (task 45, in position #5 of route #4) may have had its arrival times changed. For that constraint, column "Up to Date" is therefore changed back to 0.
Having found an infeasibility, the infeasible path identification procedure is similar to the one presented in the previous separation routine: for the route where the infeasibility was verified, the infeasible path will extend from the task where $t_i^\Gamma = a_i$ up to the task where $t_i^\Gamma > b_i$; for the remaining routes, the paths will extend from the task where $t_i^\Gamma = a_i$ up to their corresponding tasks subject to synchronisation.

Table 4.7: Illustration of the infeasible path separation routine based on multiple synchronisation constraints

#	Type	i	j	λ_{ij}	μ_{ij}	Task to update arrival times			Reference Task			Up to Date
						Task	Position	Route	Task	Position	Route	
7	LB	35	11	0.0	--	11	1	1	35	1	2	1
4	LB	45	31	0.0	--	31	1	3	45	5	4	0
5	LB	33	22	0.0	--	22	2	2	33	3	3	1
1	LB	15	42	0.0	--	42	2	4	25	5	1	1
6	UB	43	52	--	0.0	43	3	4	52	2	1	0
8	UB	14	27	--	0.0	14	4	1	27	4	2	0
2	UB	15	42	--	0.0	25	5	1	42	2	4	0
3	UB	45	31	--	0.0	45	5	4	31	1	3	0

4.3.4 Cuts construction

After finding infeasibilities through the previously described separation routines, cuts are constructed and fed to the mathematical model. The definition of these cutting planes is now provided.

4.3.4.1 Subtour elimination cuts

The subtour elimination cuts are applicable to all subtours that were found through the subtour elimination procedure described in 4.3.3.1. Considering set S containing all tasks involved in a subtour, the following cutting plane (4.68) will be added.

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \tag{4.68}$$

4.3.4.2 Capacity cuts

The capacity cuts are applied whenever infeasible paths are identified in the separation procedure based on capacity constraints, described in 4.3.3.2. For each infeasible path identified in this separation procedure, let S be the set of all tasks involved in said path. Under these conditions, the following cutting plane (4.69) will be added.

$$\sum_{i \notin S} \sum_{j \in S} \geq x_{ij} \left\lceil \frac{\sum_{i \in S} q_i}{Q} \right\rceil \tag{4.69}$$

with q_i being the demand of task i and Q being the vehicle capacity.

4.3.4.3 Infeasible path cuts

The infeasible path cuts are introduced in the branch-and-cut for each one of the infeasible paths that were identified in any of the developed separation routines – i.e., capacity in-

feasibilities, operation constraints infeasibilities and infeasibilities based on time window bound.
Given an infeasible path $R = (r_1, ..., r_{|R|})$, of length $|R|$, the infeasible path cutting plane that is introduced is defined in Equation (4.70).

$$\sum_{i=1}^{|R|-1} x_{r_i,r_{i+1}} \leq |R| - 2 \tag{4.70}$$

4.3.4.4 Sets of infeasible path cuts

The cuts for sets of infeasible paths are introduced in the branch-and-cut algorithm for each one of the sets of infeasible paths that were identified through the separation routines based on synchronisation constraints. Given a set of P infeasible paths $\mathcal{P} = \{R_1, ..., R_P\}$, each one of them with its own length $\left|R_p\right|$, the cutting plane that is introduced to the mathematical model based on this set of infeasible paths is the one presented in Equation (4.71).

$$\sum_{p=1}^{P} \sum_{i=1}^{|R_p|-1} x_{r_i^p, r_{i+1}^p} \leq \sum_{p=1}^{P} \left|R_p\right| - P - 1 \tag{4.71}$$

4.4. Computational experiments

A set of computational experiments was devised with the purpose of evaluating various indicators of the models and the solution approach. The general conditions and assumptions of the conducted experiments are described ahead.

4.4.1 Test instances and experiments parameters

The instances used in the computational experiments of this paper were based on the well-known instances from Solomon [1987], originally designed for the VRP with Time Windows. The Solomon instances are divided into three groups: 1) a group of instances whose generated customers are geographically clustered (group C); 2) instances where the locations of customers are randomly generated (group R); 3) instances where the locations of customers are both geographically clustered and randomly generated (group RC). The instances contain 100 customers. However, it is common practice to only consider a portion of first customers for less tractable problems. In the case at hand, only the first 25 customers were considered.
The adaptation of these instances to the VRP with Synchronisation was performed as follows. It was assumed that operations synchronisation will always occur at the same location and at the same time (i.e., exact synchronisation). This assumption allows us to evaluate the developed approaches for the most adverse and constraining scenario. In the process of generating the problem instances, a ratio of customers requiring operations synchronisation is previously defined, which will specify how many customers will require two vehicles to

be present at their location at the same time. The "synchronised" customers are selected randomly, and in these cases, the customer task nodes are duplicated for the location to be able to receive both vehicles. These tasks are then added to the set of synchronised operations, with $\lambda_{ij} = \mu_{ij} = 0$. The customers that were not selected for operations synchronisation remain unchanged.

The instances were generated considering a synchronisation ratio of 0% (without synchronisation, i.e., a standard VRP with Time Windows) and of 25% (i.e., with 25% of customers in need to be visited by two vehicles at the same time).

Different levels of robustness were also considered in these instances. To that effect, different combinations of budgets of uncertainty and normalised deviations were considered as in Munari et al. [2019]. The Γ values that were considered were 0 (deterministic), $1, 5$ and 10. The different combinations of deviations and budgets of uncertainty from Munari et al. [2019] were adopted, thus constituting a total of 10 combinations. Therefore, the Δ values that were considered were $0.10, 0.25, 0.50$.

The computational experiments in a machine with the Microsoft Windows 10 operating system, 32 GB of RAM and an Intel Xeon E5-2650 v2 @ 2.60 GHz CPU, with capacity for 32 simultaneous processing threads. The commercial solver used for the experiments was Gurobi Optimizer 9.0.

4.4.2 Model size analysis

With the purpose of obtaining an overview of the model sizes at hand, as well as to quantify the additional overhead from acknowledging operations synchronisation and different budgets of uncertainty, the general 3-index and the simplified 2-index formulations were instantiated for all 56 Solomon instances and all 10 combinations of Γ and Δ.

The general 3-index model was instantiated for the version without synchronisation and the simplified 2-index model was instantiated for both versions of the models without synchronisation and with exact operations synchronisation at 25%, thus corresponding to a grand total of 1,680 mathematical models.

It was observed that, under these experimental conditions, generating mathematical models with synchronisation by resorting to the 3-index formulation resulted in very large model sizes and very high consumption of computational resources, which rendered impractical the usage of these models to perform computational experiments. This observation is consistent with other reports from the literature [e.g., Agra et al., 2012, Lee et al., 2012] when using robust optimisation models with 3-index routing formulations.

The obtained models were analysed in terms of their size, whose results are summarised in Table 4.8.

The exhibited results show that the number of generated variables for the general 3-index model ranges from 13,125 to 41,175, while in the simplified 2-index model without synchronisation, this indicator ranges from 525 to 1,647, thus constituting a reduction of 96% in the number of variables. For the number of generated constraints, this percentual reduction is approximately the same, with this indicator ranging from 15,286 up to 382,526 in the 3-index model, and from 682 to 15,350 in the 2-index model.

When analysing the models containing exact synchronisation at 25% and comparing them

Table 4.8: Comparison of model sizes

Γ	Statistic	General 3-index model		Simplified 2-index model			
		w/o sync.		w/o sync.		w/ exact sync. 25%	
		# vars	# constr.	# vars	# constr.	# vars	# constr.
0	Minimum	13,125	15,826	525	682	765	1,045
	Average	26,611	29,312	1,064	1,221	1,574	1,852
	Maximum	34,425	37,126	1,377	1,534	2,077	2,357
1	Minimum	13,800	29,651	552	1,235	798	1,838
	Average	27,286	56,622	1,091	2,314	1,607	3,454
	Maximum	35,100	72,251	1,404	2,939	2,110	4,462
5	Minimum	16,500	82,351	660	3,343	930	4,882
	Average	29,986	163,265	1,199	6,580	1,738	9,736
	Maximum	37,800	210,151	1,512	8,455	2,242	12,754
10	Minimum	19,875	148,226	795	5,978	1,095	8,687
	Average	33,361	296,569	1,334	11,912	1,901	17,588
	Maximum	41,175	382,526	1,647	15,350	2,407	23,119

with their counterparts without synchronisation, we are able to quantify a percentual increase in the number of variables from 38% up to 51%. The number of constraints also increases by roughly the same order of magnitude, as the percentual increase remains practically *on par* with the increase in the number of variables, from 45% to 54%.

The results also demonstrate that, as would be expected, the number of variables increases in a fairly linear manner with the increase of the budget of uncertainty Γ, with a percentual increase of between 2 and 5 percentage points per unit of Γ that is increased. However, this is not the case for the number of constraints: in fact, when comparing the number of generated constraints between the models with $\Gamma = 1$ and $\Gamma = 5$, we quantify a percentual increase between 166% and 186%, as opposed to the percentual increase from 78% up to 81% observed when comparing the models with $\Gamma = 5$ and $\Gamma = 10$.

In conclusion, these results suggest that the 2-index model should be more easily scalable. Even when considering its version with exact synchronisation, the 2-index model is incomparably more compact than the 3-index model without synchronisation.

4.4.3 Mathematical models general performance evaluation

The developed general 3-index and simplified 2-index models were fed to the commercial solver with a maximum wall time of 3,600s. The obtained results for these runs are summarised in Tables 4.9 and 4.10. Table 4.9 shows the number of models per instance group that were able to be solved optimally and sub-optimally, as well as the average optimality gaps. Table 4.10 presents the average optimality gaps results for each deviation value and budget of uncertainty.

Table 4.9: Performance comparison between the 3-index and 2-index models

Group	No. tests	General 3-index model			Simplified 2-index model					
		w/o sync.			w/o sync.			w/ exact sync. 25%		
		# sol	# opt	Gap	# sol	# opt	Gap	# sol	# opt	Gap
C1	90	69	38	35.5%	90	90	–	90	61	14.0%
C2	80	64	10	30.8%	80	80	–	80	54	19.8%
R1	120	66	8	42.5%	108	94	8.7%	108	53	12.4%
R2	110	91	0	30.9%	110	99	3.4%	110	42	11.9%
RC1	80	56	0	29.5%	77	69	18.1%	77	16	12.7%
RC2	80	63	0	49.7%	80	42	15.1%	80	20	28.6%

Legend: Group: Solomon instance group; # sol: number of tests that were able to obtain a solution; # opt: number of tests solved to optimality; Gap: average optimality gap among the tests that were not solved to optimality. All tests were run with a maximum time limit of 3,600s.

In a general analysis, it is possible to observe that the more easily solvable instance groups are the clustered ones (groups C1 and C2), followed by the randomly generated ones (groups R1 and R2), due to the high number of instances that were able to be solved, some of them optimally. This behaviour appears to remain constant regardless of the model versions considered.

Analysing the results obtained by the general 3-index model, it is possible to verify that the solver was only able to optimally solve a total of 56 (out of 560 models) for the version without synchronisation. The results show that the commercial solver is unable to find a single solution for 151 out of the 560 3-index models. This last indicator, along with the high variability of the optimality gaps, which range from 2.5% up to 78.5%, clearly demonstrates the lack of tractability of the general 3-index model.

The simplified 2-index model, on the other hand, behaves more favourably. Out of the 560 runs, only 15 are unable to find an initial solution within the established wall time. The number of models run to optimality amounts to 474 in the model version without synchronisation and 246 in the version with exact synchronisation at 25%. The optimality gaps for the 2-index models that were not run to optimality are significantly lower than the ones for the 3-index models, although still far from ideal.

In sum, these results confirm that the 3-index is hardly scalable and, for the 2-index model, it is possible to state that the introduction of the robust component *per se* into the problem does not significantly impact the performance of the models, as an overwhelming majority of instances were solved for the model versions without synchronisation and the optimality gaps did not vary significantly. However, when introducing operations synchronisation, we observe that its increased complexity, allied with the increased complexity of budgets of uncertainty, increases the optimality gaps by about 20 percentage points, thus motivating the need for alternative methods to facilitate convergence.

Table 4.10: Dispersion of the optimality gaps for different model versions, budgets of uncertainty and deviations

Γ	Δ	General 3-index model			Simplified 2-index model					
		w/o sync.			w/o sync.			w/ exact sync. 25%		
		min	avg	max	min	avg	max	min	avg	max
0	0.00	4.1%	28.3%	61.2%	6.7%	12.6%	16.2%	2.6%	14.6%	42.4%
1	0.10	5.2%	28.4%	54.8%	7.3%	14.4%	20.7%	2.8%	14.8%	43.9%
	0.25	2.5%	31.7%	66.2%	7.3%	13.9%	18.5%	3.2%	15.8%	44.3%
	0.50	5.2%	32.2%	53.9%	4.4%	12.7%	16.8%	0.1%	15.5%	43.0%
5	0.10	8.5%	35.4%	66.8%	0.7%	10.4%	21.2%	3.0%	16.5%	45.0%
	0.25	9.1%	32.6%	74.6%	4.7%	13.2%	18.4%	2.1%	16.5%	44.7%
	0.50	3.0%	35.4%	66.9%	3.6%	14.6%	21.2%	1.5%	17.2%	45.3%
10	0.10	10.5%	42.4%	73.3%	2.0%	9.0%	22.2%	2.0%	16.6%	45.1%
	0.25	9.0%	44.0%	70.8%	0.6%	10.4%	22.5%	3.1%	17.2%	45.5%
	0.50	3.1%	43.9%	78.5%	2.7%	14.8%	32.1%	3.3%	18.7%	45.7%

4.4.4 Comparison of routing solutions

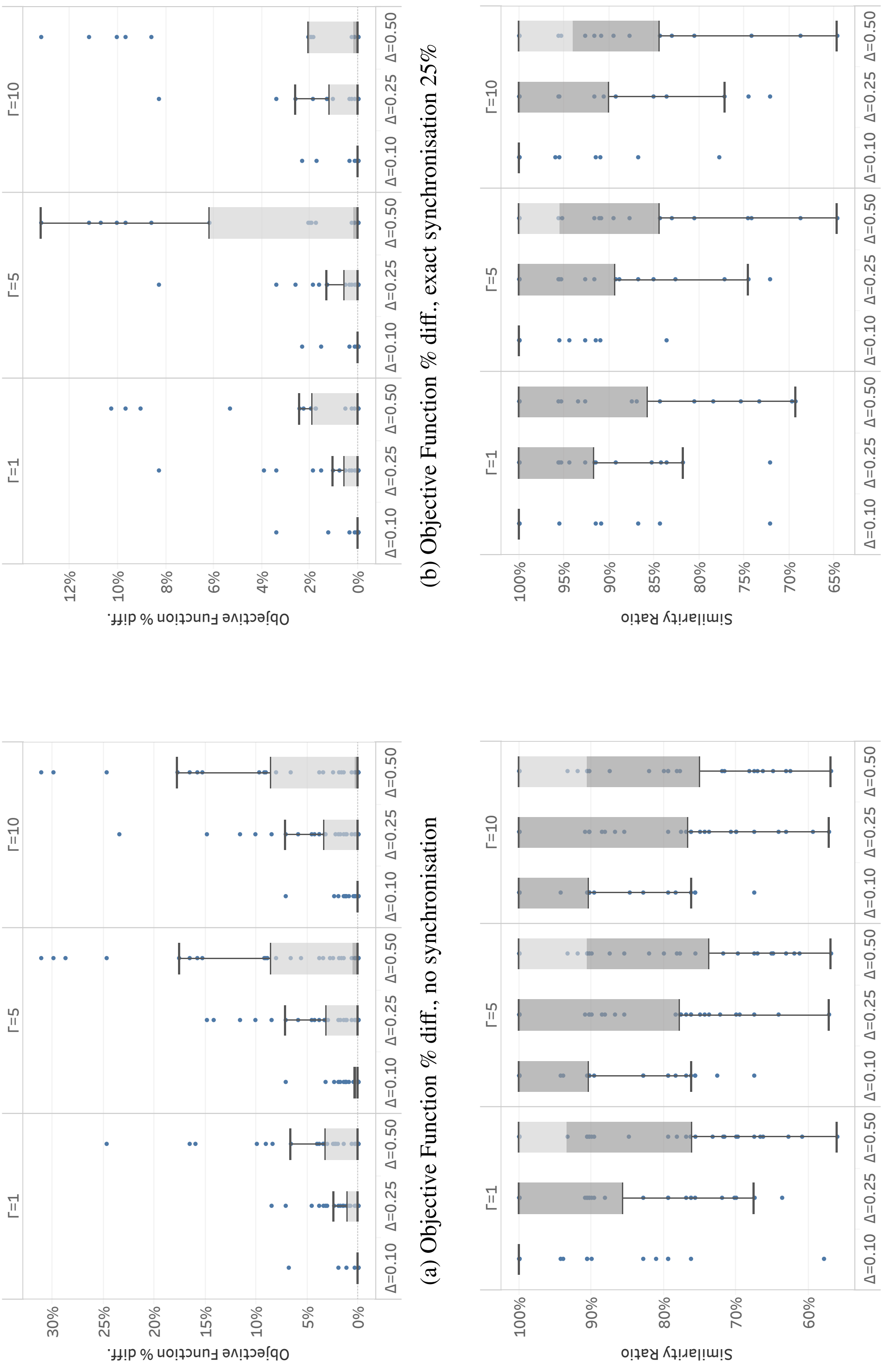
To evaluate the impact of synchronisation and robust optimisation in the obtained routing solutions, the solution structures were analysed and two indicators were calculated to enable comparisons between model runs in an aggregate manner. The first indicator consists of the percentual variation ("% diff.") of the value of the objective function between two pairs of models. The second indicator, for which the name "route similarity ratio" was adopted, consists of the ratio between the number of routing arcs that remained unchanged from one solution to another and the total number of distinct arcs that are traversed in both solutions.

These two ratios were calculated between all combinations of budgets of uncertainty, as well as deviation values and model versions.

For performing these comparisons in the most reliable way possible, only nearly-optimal solutions were considered, and only comparisons for the simplified 2-index model were calculated due to it being the most tractable model. For the purpose of this paper, a solution is nearly-optimal if its corresponding model run obtained an optimality gap that is not higher than 5.0%.

Figure 4.1 and its sub-figures present box and whiskers plots representing the dispersion of these indicators, using the deterministic models ($\Gamma = 0, \Delta = 0$) as a basis for the calculation of the indicators. Specifically, Figures 4.1a and 4.1b present the dispersion of the percentual variation of the objective function for different deviation values, while Figures 4.1c and 4.1d present the dispersion of the route similarity ratio.

In both indicators, it is possible to observe the increase in the dispersion of this ratio as the Γ and Δ values increase, as would be expected. For the % diff. indicators, most occurrences



(a) Objective Function % diff., no synchronisation

(b) Objective Function % diff., exact synchronisation 25%

(c) Route Similarity Ratio, no synchronisation

(d) Route Similarity Ratio, exact synchronisation 25%

Figure 4.1: Visual representation of the dispersion of comparison indicators (comparison basis: $\Gamma = 0, \Delta = 0$)

take values close to 0%. For the route similarity ratio, similar conclusions can be taken, as more than 50% of the occurrences of this ratio take values greater than 90%. This suggests that the impact of robust optimisation in the routing plans is highly circumstantial and dependent on the specific instance.

Nevertheless, the figures show that there is a clear increase in the dispersion of both indicators when transitioning from $\Gamma = 1$ to $\Gamma = 5$. However, there is hardly any difference between the dispersion values of $\Gamma = 5$ and $\Gamma = 10$. This suggests that the point of diminishing returns when considering increasing budgets of uncertainty must lie between $\Gamma = 1$ and $\Gamma = 5$.

Another interesting observation that can be taken from the analysis of these figures consists in the fact that the overall dispersion of the objective function values appears to decrease when introducing synchronisation constraints. While for the model versions without synchronisation the % diff. indicators may reach up to 35%, this measure decreases to about 14% for the model versions with exact operations synchronisation at 25%. For the route similarity ratio, its overall dispersion decreases slightly, from minimum values of 55% for model versions without synchronisation to minimum values of about 60% for the model versions with exact operations synchronisation at 25%.

To allow for a more direct comparison of the impacts of synchronisation in the routing plans, Figure 4.2 represents the dispersion of the route similarity ratio for the model versions with exact operations synchronisation at 25%, using as a basis for comparison the model versions without synchronisation.

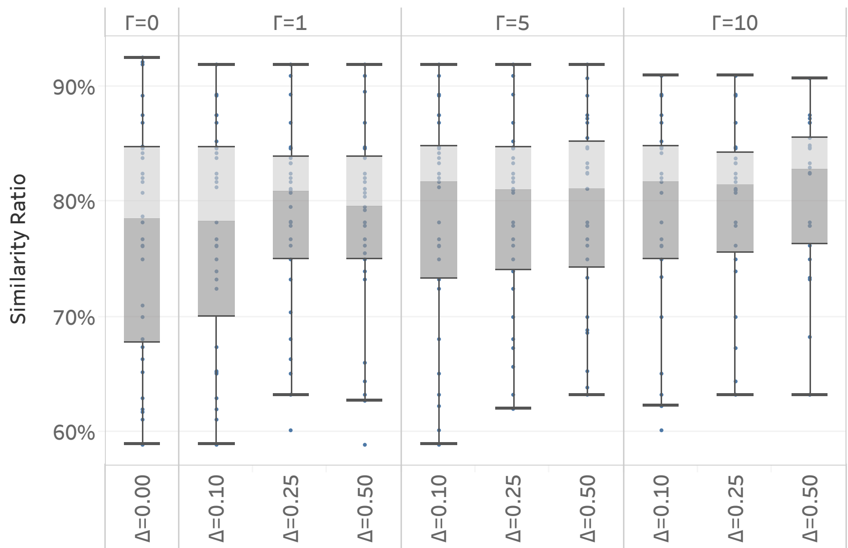


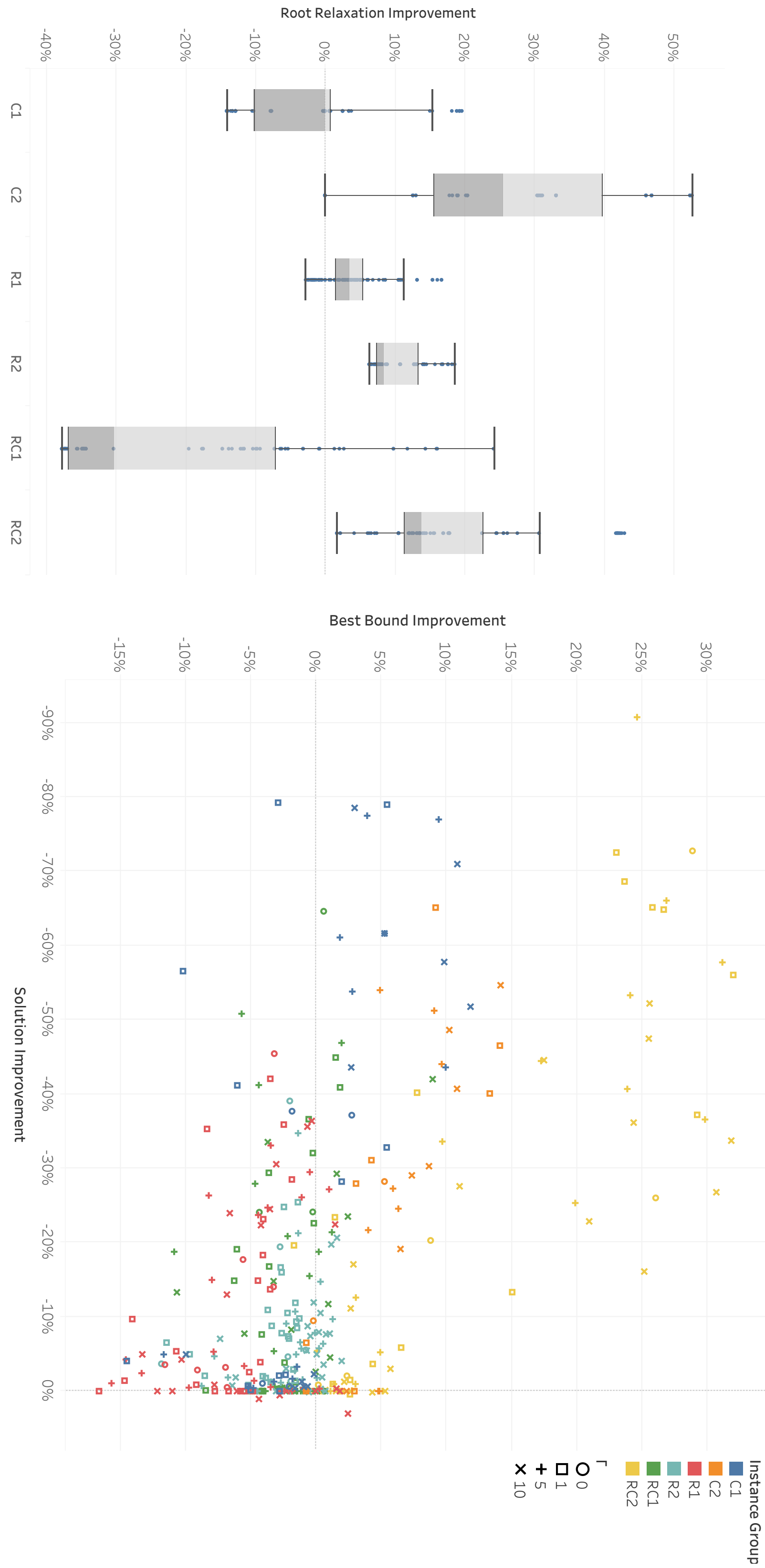
Figure 4.2: Dispersion of the route similarity ratio for exact synchronisation at 25% (comparison basis: versions without synchronisation)

The figure shows that the solutions of the models with exact synchronisation at 25% resort to at least 55% of the same arcs present in the solution from the models without synchronisation. The similarity ratio seems to decrease its variability as budgets of uncertainty and normalised deviations also increase, thus creating solutions that are apparently more

similar to the ones found in the model versions without synchronisation.
This phenomenon may occur because synchronisation constraints may induce similar effects to the ones induced by an increased level of robustness of the solution. This may be a possible explanation since the introduction of synchronisation constraints usually induces an increase in vehicle idle times, which is also an observed effect when considering worst-case travel times in the context of robust optimisation.

4.4.5 Evaluation of the branch-and-cut approach

Computational experiments were also devised to evaluate the performance of the proposed solution approach. The experiments used the 2-index model version with exact synchronisation at 25%.
The general parameters that were adopted for the solution algorithm are as follows. In the initialisation steps, the arc elimination procedure was invoked as well as the infeasible path identification procedure. In this latter procedure, and given the size of the instances, only paths of length 3 (i.e., 2-arc paths) were tested. Valid inequalities were also introduced into the models, which were generated for all of their possible combinations of set S whose cardinality is between 2 and 3. This is due to the fact that the generations of valid inequalities based on sets with higher cardinality significantly increasing the time expended for model generation. All models ran for a maximum time limit of one hour (3,600s).
The obtained results for the solution approach are visually summarised in Figure 4.3 by using the values obtained through the standalone solver approach as a basis for comparison. Figure 4.3a measures the direct impact of the initialisation procedures on the root relaxation values obtained by the solver, which are obtained before the startup of the branch-and-bound algorithm embedded in the solver. Figure 4.3b is a scatter plot that represents the percentual improvements of the final relaxation and solution values, i.e., at the end of the resolution process. To increase the legibility of this graph, instances that were able to be solved optimally in both the standalone solver approach and the branch-and-cut approach were removed.
Through the analysis of Figure 4.3a we can conclude that the initialisation procedures that were envisaged allow the solver to increase the root relaxation for approximately 75% of all instances. Analysing the dispersion of this indicator between instance groups, the percentual improvements are rather heterogeneous. The most expressive improvements are seen in instance groups C2 and RC2, which register improvements in all instances. The most consistent improvements are seen in instance group R2 since they exhibit the least dispersion. On the other hand, instance groups C1, R1 and RC1 exhibit more mixed results, with some instances improving the root relaxation and others having it worsen. For instance group C1, about half of the instances register improvements, while for instance group R1 less than a quarter of the instance have registered a worse root relaxation. Instances of group RC1 exhibit the worst results, with over 75% of the instances obtaining worse root relaxation values. In sum, these results suggest that the initialisation procedures have a potentially positive effect on the strengthening of the root relaxation of most models, although this behaviour appears to be preponderant in instance groups C2, R2 and RC2.
Now proceeding to the analysis of Figure 4.3b, which compares both the model's final



(a) Percentual improvement of the root relaxation

(b) Percentual improvement of the best bound and final solution values

Figure 4.3: Comparison of performance indicators of the branch-and-cut approach (comparison basis: mathematical solver approach)

linear relaxation and solution improvements, we are able to verify that the majority of the plot points are situated in the third quadrant, meaning that most model instances obtained both worse solution and linear relaxation values with the branch-and-cut approach. The instances that can obtain better solutions with the branch-and-cut approach are very few. However, a substantial number of instances can reach the same solution value as the standalone solver. It may be possible that these instances were already able to reach optimal solutions, although without its optimality proven. Instances from group RC2 exhibit the most expressive improvements in the linear relaxation values, although at the expense of much worse solution values. Worse solution values also appear to be a consistent behaviour occurring in instances of groups C1 and C2. In fact, there appears to be a correlation between the linear relaxation improvements and the root relaxation improvements previously observed in Figure 4.3a. The significant root relaxation improvements of instances of groups C2 and RC2 appear to be attenuated throughout the solution process. Also, taking into account the information of Figure 4.3b, the budget of uncertainty Γ does not seem to have an impact on the quality of the obtained results. In sum, the obtained results hint that further tuning of the branch-and-cut algorithm may be necessary. Although a subset of the instances appear to have their linear relaxations strengthened, this did not translate into algorithm convergence for obtaining better solutions.

4.5. Conclusions and future work

The work presented in this paper is one of the first studies of a Vehicle Routing Problem subject to travel time uncertainty and synchronisation constraints. More precisely, we consider a robust optimisation approach based on the budget of uncertainty concept devised by Bertsimas and Sim [2004]. This approach, which has been popularized in the literature, allows for finer control of the degree of conservatism that the decision agent may want to incur in their planning activities.

Although robust optimisation is typically known for being more tractable than alternative approaches, such as stochastic optimisation, modelling and embedding robustness into a mathematical model may not be trivial, depending on the specific source of uncertainty being considered. Considering travel times as the source of uncertainty for this problem raised a few challenges in this regard. The approach proposed by Munari et al. [2019], upon which this paper was built, solved part of the problem, specifically concerning the modelling of arrival time variables, by introducing an additional index into these variables, γ, which represents the number of worst-case travel times that have occurred thus far in a route. However, when considering synchronisation, because the worst-case occurrences are not known ahead of time, this requires the mathematical model to have to introduce synchronisation constraints for variables of all possible γ combinations between routes, thus leading to a substantial increase in the number of constraints involving synchronised tasks.

Two mathematical models were envisaged for this problem. The general 3-index model is a more flexible, although less scalable, formulation. The computational experiments showed that this formulation generates large-size and hardly tractable models, especially for instances with a significant number of synchronisation constraints. The use case for this model should, therefore, apply to more specific, smaller instances that require the higher degree of flexibility that this formulation enables. On the other hand, the simplified 2-index model should be used when possible, since its scalability is incomparably better than the general model.

The differences in model scalability have an impact on model performance. The simplified 2-index model consistently outperforms the general model due to its lower number of variables and constraints, although a significant number of instances remained unsolved to optimality.

The branch-and-cut algorithm that was developed intended to significantly reduce the large optimality gaps that were still observed in most instances. To that effect, several initialisation procedures, valid inequalities and cutting planes were implemented to facilitate convergence and reduce initial solver overhead by removing the most complex constraints from the initial model. However, the obtained results do not suggest that this algorithm has provided consistent improvements when compared to a standalone solver approach. It is not easy to explain this behaviour, but a possible cause for these results may reside in the fact that the solver may be able to internally infer some cuts by itself when the entire model is fed to it. On the other hand, when the branch-and-cut algorithm is invoked, this is not possible to occur since the initial model that is fed to the solver is stripped from essential constraints.

The results from this research have raised some significant research opportunities. First, the results indicate that further work is needed in the development of a solution approach for this problem. Within the scope of branch-and-cut algorithms, further work could be envisaged in the development of cutting planes or valid inequalities that could accelerate convergence. However, other approximate solution methods may also be feasible to develop, such as heuristic or matheuristic approaches.

Secondly, one of the assumptions of this paper consisted in considering that each route contained an independent budget of uncertainty, thus meaning that each route should be feasible when Γ worst-case travel times occured, regardless of which arc they occured. However, in a more realistic setting, it would be probably be worthwhile to consider a global budget of uncertainty that could be evenly or unevenly distributed throughout routes. In a preliminary analysis, the modelling challenges behind this assumption are not negligible, but this challenge would probably be scientifically impactful if it were to be tackled.

Acknowledgements

The first author acknowledges the support of the Ph.D. Grant SFRH/BD/136314/2018, awarded by the Portuguese Science and Technology Foundation (FCT) and financed by the European Social Fund (ESF) and by National Funds of the Portuguese Ministry of Science, Technology and Higher Education (MCTES) through the Human Capital Operational Programme (POCH).

Bibliography

Agostinho Agra, Marielle Christiansen, Rosa Figueiredo, Lars Magnus Hvattum, Michael Poss, and Cristina Requejo. Layered formulation for the robust vehicle routing problem with time windows. In *Lecture Notes in Computer Science*, pages 249–260. Springer Berlin Heidelberg, 2012. doi: 10.1007/978-3-642-32147-4_23.

Igor Averbakh. On the complexity of a class of combinatorial optimization problems with uncertainty. *Mathematical Programming*, 90(2):263–272, April 2001. doi: 10.1007/pl00011424.

Dimitris Bertsimas and Melvyn Sim. The price of robustness. *Operations Research*, 52(1): 35–53, February 2004. doi: 10.1287/opre.1030.0065.

John R. Birge and François Louveaux. Basic properties and theory. In *Introduction to Stochastic Programming*, pages 103–161. Springer New York, 2011. doi: 10.1007/978-1-4614-0237-4_3.

N. Furian, M. O'Sullivan, C. Walker, and S. Vössner. Evaluating the impact of optimization algorithms for patient transits dispatching using discrete event simulation. *Operations Research for Health Care*, 19:134–155, 2018. doi: 10.1016/j.orhc.2018.03.008.

C Lee, K Lee, and S Park. Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *Journal of the Operational Research Society*, 63(9):1294–1306, September 2012. doi: 10.1057/jors.2011.136.

Pedro Munari, Alfredo Moreno, Jonathan De La Vega, Douglas Alem, Jacek Gondzio, and Reinaldo Morabito. The robust vehicle routing problem with time windows: Compact formulation and branch-price-and-cut method. *Transportation Science*, 53(4):1043–1066, July 2019. doi: 10.1287/trsc.2018.0886.

Fernando Ordóñez. Robust vehicle routing. In *Risk and Optimization in an Uncertain World*, pages 153–178. INFORMS, September 2010. doi: 10.1287/educ.1100.0078.

Sophie N. Parragh and Karl F. Doerner. Solving routing problems with pairwise synchronization constraints. *Central European Journal of Operations Research*, 26(2):443–464, 2018. doi: 10.1007/s10100-018-0520-4.

Ricardo Soares, Alexandra Marques, Pedro Amorim, and Sophie Parragh. Synchronisation in vehicle routing: a literature-based classification schema and modelling framework. Submitted for publication, 2022.

Marius M Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.*, 35(2):254–265, apr 1987. doi: 10.1287/opre.35.2.254.

Chapter 5

Conclusions and future work

This thesis approaches the topic of Synchronisation in Vehicle Routing Problems (VRPs) from different perspectives, being grounded on three relevant research questions.

The first research question is addressed in Chapter 2. It takes upon itself the conceptualisation and systematisation of concepts and problem aspects present in the VRP with Synchronisation. The relevance of this research question is motivated by the lack of a homogeneous framework of synchronisation in the literature. The scientific contributions of this Chapter successfully answer Research Question 1 through the integrative literature review that is conducted. The chapter provides an updated state of the art on VRPs with Synchronisation and clarifies the concept of synchronisation in the context of vehicle routing. A general modelling framework for VRPs with Synchronisation is also proposed as an attempt to define a general class of VRPs with Synchronisation.

Although this chapter appropriately tackles its research question, routing problems with synchronisation still entail several challenges. It is expected that solution approaches for VRPs with Synchronisation continue to be developed since the instance sizes that current approaches can handle are still limited. This is a consequence of the $\mathcal{NP}$-hard nature of the standard VRP, which is a particularisation of the VRP with Synchronisation. It is also expected that further applications of routing with synchronisation will appear as this topic matures. Routing problems considering the synchronisation between manned and unmanned vehicles will probably be one of the types of problems that will see more research being done, mainly due to the emerging trends in vehicle routing with drones and autonomous vehicles.

While Research Question 1 has a theoretical focus, the other two research questions of this thesis are focused on more practical and specific challenges that arise in VRPs with Synchronisation.

The second research question of this thesis is addressed in Chapter 3. The purpose of this research question is to present an approach to tackle and solve an operational routing problem involving multiple interlinked synchronisation constraints occurring between multiple types of vehicles. Based on a case-study from the biomass sector, the intricacies of the problem requirements present an innovative application of synchronisation that further advances the state of the art in this field. The research question is answered through the proposal of a compact mathematical formulation and the development of a matheuristic solution approach. The scientific outputs are then leveraged to obtain managerial insights for the biomass sector. Given the advantages of synchronisation, it is expected that new applications of synchronisation involving the integration of different processes of the supply chain emerge as solution algorithms also progress.

The third research question of this thesis is addressed in Chapter 4. This question aims to tackle parameter uncertainty in VRPs with Synchronisation, which is unexplored in the literature. The research question is answered through the development of a robust optimisation approach grounded on mathematical formulations that model uncertainty in travel times between customers. The proposed approach is based on previously consolidated methodologies in robust optimisation, which are adapted to the case at hand. An exact approach, namely a branch-and-cut algorithm, is developed and tested. A set of valid inequalities and cutting planes is developed based on problem specificities of the VRP with Synchronisation, which may serve as a basis for future tests and research in this field. The scientific results of this research suggest that further work must be envisaged in VRPs with Synchronisation subject to uncertainty, specifically in the realm of solution approaches. Nevertheless, since this topic is unexplored in the literature, alternative sources of uncertainty could also be tackled.

Considering that a PhD thesis shall reflect the PhD candidate's training within the scope of the doctoral programme, this thesis contains two additional chapters in the form of appendices. Each of the appendices contains research papers where the PhD candidate applies his competencies in Operations Research (OR) towards the development of impactful research. Appendix A researches the topic of network design for logistics planning. It uses mathematical programming to solve a complex assignment problem in the biomass sector, where biomass flows from the roadside of forest piles must be optimally assigned to their consumers. To that effect, several operational requirements in terms of workforce, transportation and biomass processing constraints are considered. The research is leveraged to optimise a set of operational scenarios that provide managerial insights to the decision agent.

Appendix B consists of research work on the topic of integrated routing. It studies the possible integration of inbound and outbound logistics in the context of a case-study in the wood-based panel industry. Several OR techniques are adopted by the PhD candidate to tackle this challenge. Mathematical models are constructed for each of the conjectured integration strategies. A solution method is devised with the purpose of obtaining greater quality solutions for larger problem instances. Several computational experiments are also devised to weigh the pros and cons of each of the possible integration strategies.

It is believed that the scientific results exhibited in this thesis demonstrate thorough, critical and impactful research work that satisfies the demanding standards of the Faculty of Engineering of the University of Porto.

Therefore, the PhD candidate respectfully submits this thesis to the jury in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering and Industrial Management.

Appendix A

Network design for transportation planning with synchronisation

Planning woody biomass supply in hot systems under variable chips energy content

Alexandra Marques* · **Jussi Rasinmäki**‡ · **Ricardo Soares** *† · **Pedro Amorim***†

Published in *Biomass and Bioenergy*, 2018. DOI: 10.1016/j.biombioe.2017.11.016

*INESC TEC - Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Porto, Portugal
†Faculdade de Engenharia, Universidade do Porto, Porto, Portugal
‡Simosol Oy, Riihimäki, Finland

Biomass and Bioenergy 108 (2018) 265–277

Contents lists available at ScienceDirect

Biomass and Bioenergy

journal homepage: www.elsevier.com/locate/biombioe

Research paper

Planning woody biomass supply in hot systems under variable chips energy content

Alexandra Marques[a,*], Jussi Rasinmäki[b], Ricardo Soares[a], Pedro Amorim[a]

[a] INESC TEC Technology and Science, Campus da FEUP, 4200 - 465 Porto, Portugal
[b] Simosol Oy, Rautatietori 4, 11130 Riihimäki, Finland

ARTICLE INFO

Keywords:
Biomass supply chain planning
Forest residues
Synchronization of chipping and transportation
Moisture content
Energy content
Mathematical programming

ABSTRACT

The growing economic importance of the biomass-for-bioenergy in Europe motivates research on biomass supply chain design and planning. The temporally and geographically fragmented availability of woody biomass makes it particularly relevant to find cost-effective solutions for biomass production, storage and transportation up to the consumption facility. This paper addresses tactical decisions related with optimal allocation of wood chips from forest residues at forest sites to terminals and power plants. The emphasis is on a "hot-system" with synchronized chipping and chips transportation at the roadside. Thus, decisions related with the assignment of chippers to forest sites are also considered. We extend existing studies by considering the impact of the wood chips energy content variation in the logistics planning. This is a key issue in biomass-for-bioenergy supply chains. The higher the moisture content of wood chips, the lower its net caloric value and therefore, a larger amount of chips is needed to meet the contracted demand. We propose a Mixed Integer Programming (MIP) model to solve this problem to optimality. Results of applying the model in a biomass supply chain case in Finland are presented. Results suggest that a 20% improvement in the supplier profit can be obtained with the proposed approach when compared with a baseline situation that relies on empirical estimates for a fixed and known moisture content in the end of an obliged storage age.

1. Introduction

Design and planning of biomass-for-energy supply chains (BESC) has been widely studied, as society reinforces the major role of biomass as a global primary energy source. In the case of woody biomass (produced from branches and other by-products of forestry operations), as in other forms of biomass (e.g. residues from agriculture, forestry, fisheries and municipal waste), the availability is temporally and geographically fragmented, which makes it particularly relevant to find cost-effective solutions for biomass production, storage and transportation up to the consumption facility (e.g. Ref. [17]).

In this paper, the company in focus is a biomass supplier that buys the forest residues from forest owners (suppliers) and delivers the wood chips to power plants (customers) in order to meet their contracted demand of energy content, expressed in terms of MWh. The sequence of operations that are responsibility of the company are: 1) Logging, ie., tree felling, delimbing the trunk and cross-cutting into pre-defined lengths with specialized harvesters or manual harvesting with chainsaws; 2) Forwarding the logs and residues with skidders, forwarders or other types of tractors from the logging site up to pre-defined stacking locations at the roadside; 3) Chipping forest residues into smaller size wood chips, with specialized chippers located at the roadside or in terminals for longer term storage; 4) Transporting forest residues or wood chips by truck from the forest sites; and finally 5) Temporary storing and drying the residues and/or chips at the roadside or in terminals. Drying usually occurs under favorable sun and wind open-air conditions, but technical drying systems can be used in terminals, with addition of heat and with forced ventilation in order to reach much lower moisture content levels.

This research focus on planning chipping, transportation and storage operations, especially during the heating season when the power plants are operating. The emphasis is on "hot systems" where wood chipping and transportation operations are synchronized at the roadside. In this case, the trailer-mounted chipper feeds directly a chargeable container mounted in the truck, which will transport the chips ultimately to the plants. The company main decisions with respect to chipping are: *1) when and where to produce the wood chips, to match wood chips availability and plants demand; 2) which chipper to assign to forest residues piles at the forest site.* Main decisions with respect to transportation are: *1) amounts from where, to where, when, what product (flows);*

* Corresponding author.
E-mail address: alexandra.s.marques@inesctec.pt (A. Marques).

https://doi.org/10.1016/j.biombioe.2017.11.016
Received 5 July 2017; Received in revised form 18 November 2017; Accepted 21 November 2017
Available online 01 December 2017

2) transportation capacity needed in each period. In respect to storage, the company main decision is: *3) how long to store/dry forest residues/wood chips and where (roadside or terminals)?* It is noteworthy that, in case of hot systems, there is no intermediate storage of wood chips between these operations, but there is usually storage of forest residues at the logging sites because chipping is done some time after harvesting. Contrarily, the “cold system” encompasses the transport of forest residues to the terminals for later chipping and storage (e.g. Ref. [11]).

The literature shows several examples of mathematical programming techniques to help plan chipping and transportation operations with the aim to minimize the cost per kWh generated (e.g. Refs. [4,6,30]). Previous research from Ref. [19] addressed the case of a large Swedish biomass-supplying entrepreneur. They developed a model to decide when and where forest residues have to be converted into chips, transported and stored in order to satisfy the contracted demand at the sawmill at a minimum cost. They assume that harvesting (and chipping) in each stand occurs in a single period and do not address the assignment of machinery to these operations. Continuous variables determine biomass flows from harvest sites and sawmills to heating plants in each time period and binary variables determine whether forest residues are forwarded or chipped, whether a sawmill has been contracted and a terminal is used in a certain time period (one month) over a planning horizon of one year. The problem was solved with a heuristic approach and applied in six scenarios of possible variations in the supply chain design. In a similar context [14], apply a MIP model to the optimization of inventory planning at the terminals in order to support the choice of chipping technology and location and the route to the heating plants. The model was implemented in a Decision Support Tool called FuelOpt. [23] and [18] also studied the biomass supply in case studies in Austria. The former proposed a Linear Programing model while the latter applies a simple stepwise heuristic approach based on the calculation of available regional forest fuel potential.

Despite these relevant efforts, the dependency between chipping and transportation operations that characterize “hot systems” is still poorly addressed. Previous research (e.g. Refs. [11,12,24]) develop simulation-based approaches to assess productivity issues related with alternative chipping systems as well as to show the importance of balancing chipping and transportation capacity to avoid unnecessary costs related with the trucks waiting time and chippers idle time. In a case similar to this [3], proposes a discrete-event simulation model to find optimal set-ups for the supply chain of crushed material, made from stumps at different road transport distances. Yet, optimization models for jointly planning chipping and transportation remain undone.

Moreover, the impact of wood chips moisture in storage and logistics planning is not yet properly addressed, although it is a key aspect of the business. Usually companies use the chips with lower moisture content as possible, because this corresponds to a higher energy content, meaning that less energy is spent to vapor the water in the wood instead of heating. Moisture content also affects negatively the efficiency of combustion (higher emissions of carbon monoxide, hydrocarbon and fine particles), increases the risk of decay during storage, and increases the transportation costs [27]. Chips moisture content is higher just after harvesting and tends to decrease along the time spent in storage (e.g. Refs. [20,21,28]. Yet, the drying rate depends on the initial moisture content, the weather conditions (specially sun and wind) during the drying period, the drying capacity of the wood, phytosanitary conditions, pile cover type and arrangement, and other features of the storage yard (e.g. dimension, soil drainage capacity) [27].

One of the few studies addressing the impact of moisture content variation in logistics planning was done by Ref. [8]. They apply a simulation model built with a state-task-network approach. Another study by Ref. [31] proposes a ‘stochastic programming-robust optimization’ model to tackle biomass supply planning, addressing uncertainty in biomass quality and biomass availability. [16] quantifies the impact of incorporating terminals between harvest locations and biorefineries. In this analysis, the decrease in moisture content is one of the considered factors for evaluating the potential benefits of intermediate storage locations. Nevertheless, most planning models for biomass supply fail to effectively capture the impact of the changes in the product properties according to storage time, and do not incorporate the storage age into the model.

The main contributions of this paper are to formulate and solve the tactical biomass supply planning problem, thus extending the work of [14,19] by explicitly considering the variation in chips energy content (or moisture content) over time in storage. Furthermore, it addresses the dependency between chipping and transportation at the roadside that characterize the “hot systems” as well as the space-time continuity of chipping operations.

The remainder of this paper is as follows. Chapter 2 presents the problem description, with emphasis on identifying the impact of the variation of the wood chips energy content in the logistics planning as well as explaining the dependencies between chipping and transportation operations that characterize the hot systems. Chapter 3 presents the proposed modelling approach. It further discusses possible variations to the general Mixed-Integer Programming model for cases where the storage age is not dealt with or the movements of the chippers between piles can be simplified. Chapter 4 presents the computational experiments for a case study of a biomass supply company in Finland. Finally, chapter 5 presents the concluding remarks.

2. Problem description

The woody biomass supply planning problem in hot systems under variable chips energy content can be formulated as follows. Considering a set of power plants (M) with a given demand of energy content (MWh) per week, the problem consists in determine 1) which piles (P) of forest residues should be chipped according to its availability and moisture content; 2) by which chippers (K), and 3) where to transport the chips, considering the possibility to use forest sites and terminals (O) for temporary storage (Fig. 1). The objective is to maximize the operational net profit, considering the revenue from wood chips sales to the plants as well as the costs of chipping, transportation and storage. This is a multi-period flow problem, where the planning periods be half-a-day, one day or even week, and the planning horizon can range from 1 to up to 6 months, the latter corresponding to the expected duration of the heating season, when the power plants are operating.

2.1. Incorporating wood chips energy content variation in logistics planning

The energy demand at the power plants is specified in MWh. This corresponds to the minimum supply during the entire cold season, when the plant is operating, while the maximum supply can be approximated by the plants processing capacity. The price per MWh vary from plant to plant and the supplier was no control over pricing, which is assumed to be fixed within the planning horizon. Depending on the

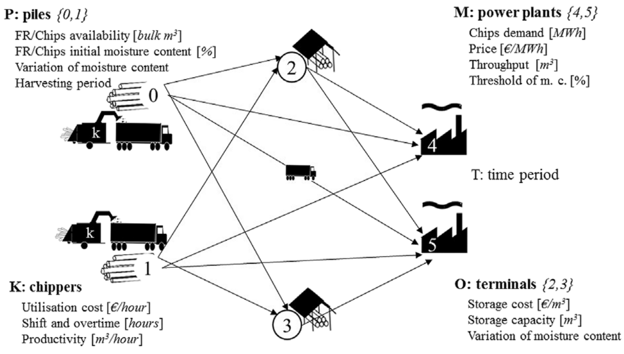


Fig. 1. Graphical representation of the biomass supply planning problem.

type of boilers installed, some plants also define thresholds in respect to the minimum energy content accepted. But in general, the larger the plant the more tolerant it is to variations in the fuel properties.

Forest residues and wood chips availability is specified in bulk m^3. This is the unit of measurement for small pieces of loose wood (e.g. wood chips, sawdust, wood pieces) that attain a total volume of one cubic meter including air gaps [25]. 1 bulk m^3 wood chips corresponds to 0.33 m^3 round wood equivalent. The piles of forest residues become available at the roadside since the period when harvesting occurred, which is known beforehand. Also the amount of forest residues available in each pile is known and is usually estimated as a percentage of the total stand volume, which can be predicted with yield and growth models based on forest inventory data.

The conversion MWh to m^3 is the wood chips energy content or its net caloric value or net heating value (ε). It corresponds to the usable heating volume released in complete burning of a specific volume of fuel, after subtracting the heat of vaporization of the water vapor (2.44 MJ per kg of water). It is computed mainly as a function of the moisture content (ϑ) (Equation (1)); other parameters are the net caloric value of oven-dry wood (ε_0) (18.5 MJ/kg) and bulk density (ρ) (kg/bulk m^3), whose values for the main tree species and moisture content can be found in the Wood Fuels Handbook [25]:

$$\varepsilon = \left(\frac{\varepsilon_0(100-\vartheta)-2.44\vartheta}{100}\right)0.278\ .\ \rho\ \ (\text{KWh/m}^3) \tag{1}$$

Consequently, the wood chips moisture content is the key parameter for the business. The higher the moisture content the higher the volume necessary to meet the demand. Thus, it is measured often in the course of biomass supply processes with portable measuring devices. The wood moisture content (or water content, or moisture content percentage on green basis (ϑ)) is the mass of water present in relation to the mass of fresh wood. $\vartheta = \frac{W_w - W_0}{W_w} \times 100$, where W_w is the wet weight of wood and W_0 is the oven-dry weight of wood. Note that some portable devices may measure the wood humidity (or moisture content on oven-dry basis (θ)), corresponding to the ratio between the mass of water present and the mass of oven-dry wood. In those situations the following conversion formula can be used $\vartheta = \frac{100\theta}{100+\theta}$. As rules of thumb in the literature (e.g. Ref. [25]), newly-chopped fresh wood has half of water and half of wood substance ($\theta = 100\%$, $\vartheta = 50\%$). Fresh wood chips have ϑ between 45% and 55%. After drying for a couple of months under favorable open air conditions it lowers to 25–40%, while in case of technical drying can reach below 20%. [27] and [15]. According to [25], the net caloric value of wood chips is around 3.4 KWh/kg (or MWh/ton).

The variation of moisture content of forest residues/wood chips along the time spent in open-air drying at the forest sites or terminals can be estimated by means of local field tests, drying curves or mathematical models or even other alternative empirical approaches. Experimental field tests help to determine the key factors impacting in the drying process (e.g. Refs. [5,28,29]). As an example [20], concluded that the key impacting factors in spruce piles (forest residues and wood chips) in Central European conditions are season, storage duration, assortment and fleece cover. In winter period (November to April), wood chips moisture content varied from 56% to 53% (average of 0,15% per week), while in summer (May to October) varied from 48% to 34% (average of 0,7% per week). Forest residues dried faster than chips. Piles covered with fleece also dried much faster due to heat accumulation and lower heat dissipation.

Graphical drying curves help to predict the expected moisture content along the time spent drying under case-specific conditions (e.g. Ref. [26]. As an example [21], proposed drying curves for spruce and beech piles in central European conditions, starting in the winter season. The maximum moisture content is reached in the beginning of the storage period and then decreases during the year until reaching the minimum moisture in summer. After this period, the wood moisture content may increase again due to rain and increased air humidity. Existing mathematical models usually predict change in moisture content as a function of mean air temperature and relative moisture. Still other studies propose more general mathematical models that predict moisture content change as a function of mean air temperature and relative moisture (e.g. Refs. [10,32]).

This research considers all these methods equally valid and can be adopted by the biomass company for the purpose of biomass supply planning at the tactical level. If fact, this research is built under two main assumptions: 1) Existing stock (and moisture content) at the beginning of the planning period is known; 2) The variation of moisture content along the time spent in storage is also known in advance for each storage location (piles and terminals), by means of local field tests, drying curves or mathematical models or even other alternative empirical approaches.

In cases where no mathematical model is available, a generic logistic function can be used (see Equation (2)) where $\alpha > 0$ is a parameter that defines the sigmoid curve's steepness, π sets the sigmoid curve's midpoint in the t axis, ϑ_0 is the initially measured wood moisture content and ϑ_{eq} is the curve's horizontal lower asymptote and may be estimated by the lowest moisture content measured in a one-year period. As an example, for the spruce piles stored in Freising, Germany during 2 consecutive years (Dec. 2002 to Nov. 2004) presented in Ref. [21], $\alpha = 0.9$ and $\pi = 4.6$.

$$\vartheta(t) = \vartheta_{eq} + \frac{\vartheta_0 - \vartheta_{eq}}{1 + e^{\alpha(t-\pi)}} \tag{2}$$

2.2. Addressing the dependency between chipping and transport at the roadside (hot system) in biomass supply planning

Chipping and transportation operations have a start-to-start dependency, meaning that both need to be present and available at the roadside so that both operations can be performed. Similarly, there is a finish-to-finish dependency. This is because forest residues are chipped directly into the trucks' container and transported just after the container is full (i.e. in the same period). Consequently, biomass supply planning should seek for an optimal balance between productivity/capacity of chippers and trucks, in order to avoid unnecessary costs related with the trucks waiting time and chippers idle time.

There are other relevant business rules related with chipping that need to be taken into account for biomass supply planning. The first is that the chipper processes one pile at a time. The chipper productivity in terms of m^3/hour depends on the type of chipper and the pile characteristics (e.g. size of the wood fuels) [25]. Low power chippers (engine ~50 kW), usually installed on the rear three point hitch of a tractor or on a trailer, only processing small diameters (up to 20 cm) and chipping productivity below 10 bulk m^3 of wood chips per hour; medium power chippers (engine 50–110 kW), usually trailer-mounted, can chip diameters up to 30 cm and the chipping productivity is up to 50 bulk m^3 of wood chips per hour; high power chippers (engine > 110 kW) installed in trailers or trucks, can chip large diameters (> 30 cm) and produce more than 100 bulk m^3 of wood chips per hour. Once chipping starts (and a truck is available) the productivity is assumed constant for that chipper in that pile. There is a daily cost of having a chipper assigned to a pile and a hourly utilization cost, that varies according to the type of equipment and ownership but it is the same either it is working or paused because no truck is available for loading. Hourly costs are higher when chipping occurs beyond the number of working hours of a regular shift. It is not mandatory that all piles are chipped within the time horizon.

Other important aspects that condition the assignment of chippers to piles are here called **space-time continuity constraints**. Temporal continuity means that the chipper remains in a pile until all the material is transported. Consequently, chipping may extend over several consecutive time periods. This is the same as saying that chipping operations cannot be interrupted once started, known in the production scheduling literature as a non-preemptive requirement (e.g Ref. [22]). Spatial continuity relates to the fact that for trailer-mounted chippers, there is a second type of truck needed to move the chipper to and from the pile, with extra operational costs. Consequently, chippers should be moved to a nearby pile to avoiding unnecessary chipper's transportation costs (and unproductive time).

In respect to transportation, this research assumes that there is an homogenous fleet of available trucks, with a coupled trailer with sidewalls or container. The loading capacity of each truck is around 87 bulk m3 (21ton). Trucks' usage cost also varies with ownership. The supplier preferably uses their own chipper and trucks. If the company does not have enough chippers and/or trucks to comply with the power plants' energy demands, then the company is able to subcontract chippers.

3. Problem modelling

The proposed modelling approach for the biomass supply chain problem builds on the MIP model developed by Ref. [19] and extends it according to the problem description. The assignment variables are extended to address hot systems. x_{kpt} take value 1 if chipping and transportation operations will occur in pile p in period t by chipper k, and 0 otherwise. As in Ref. [19], continuous variables represent the biomass flows. The linking constraints between the assignment variables and the flow variables assure that flow only exists if and when the operations are performed, therefore implementing the dependency between chipping and transportation that characterize the hot systems.

New auxiliary binary variables are added for modelling chipping movements. $z_{kp_1p_2t}$ take value 1 if there is movement of chipper k from p_1 to p_2 in the end of period t. In respect to spatial continuity, a feasible movement of the chipper between piles p_1 and p_2 requires that $z_{p_1p_2(t-1)}$ take value 1 if $x_{p_1(t-1)} = 1$ and $x_{p_2t} = 1$. Therefore, $z_{kp_1p_2(t-1)} \geq x_{kp_1(t-1)} + x_{kp_2t} - 1$. For each pile p_1 it can happen at most once along the entire planning period ($\sum_t \sum_{p2} z_{p_1p_2t} \leq 1,\ \forall\ p_1$). To assure flow connectivity, it is necessary a new set of constraints that balance the inflows and the outflows for each pile-time period, as explained in 3.2. In respect to temporal continuity, the previous constraints are sufficient to force the chipper to remain in pile p_1 in consecutive periods whenever needed, because 1) the chipper either continues in that pile p_1 or moves to another pile p_2: ($p_2 \in P \cup \{depot\};\ p_2 \neq p_1$); 2) can only move from p_1 to another pile p_2 once i.e. cannot came back to p_1 to complete the task and them move again from p_1 with minimum costs (see Fig. 2).

New continuous decision variables account for machinery/crew total number of working hours, including the shift duration and the overtime work, i.e. beyond the regular working shift duration. There variables are instrumental for dealing with the pool of chippers with heterogeneous productivity in terms of m^3/hour. Additional continuous variables h^*_{kpt} account for the amount of overtime chipping work. In cases where there is still some minor amounts left to chip in the end of a working shift in t, these variables are instrumental for modelling the trade-off between concluding chipping in that period t with additional overtime costs or delaying operations to $t+1$ with additional costs related with the chipper daily utilization (see Fig. 2).

The continuous variables representing biomass flows and stock need to be adapted to take into account the variation of moisture content according to the storage age. Previous work in logistics planning dealing with perishability in food goods (e.g. Ref. [2]) use decision variables $w^e_{u,t}$ to define the initial inventory of product u, with age e available at period t, $a = 0, \ldots \min\ \{a_u - 1;\ t - 1\}$, where a_u is the shelf-life duration of product u, right after being produced. The inventory balance constraints account for the spoilage rate for each product. This research proposes a similar modelling approach that considers a set of moisture content classes ($e \in E$). A set of continuous variables $f_{ijet_1t_2}$ represent the amount of wood chips with moisture content class e transported from supply point $i \in I$ to demand point $j \in J$ in period $t_2 \in T$, that arrived to i in period $t_1 \in T^{\varnothing}$: $\{t|t \in T,\ t < t_2\}$ (m^3). The period between t_1 and t_2 corresponds to the storing age. Similarly, variables $s_{oet_1t_2}$ are the amount of wood chips with moisture class e stored in supply point $o \in O$ in period t_2 that arrived in period t_1.

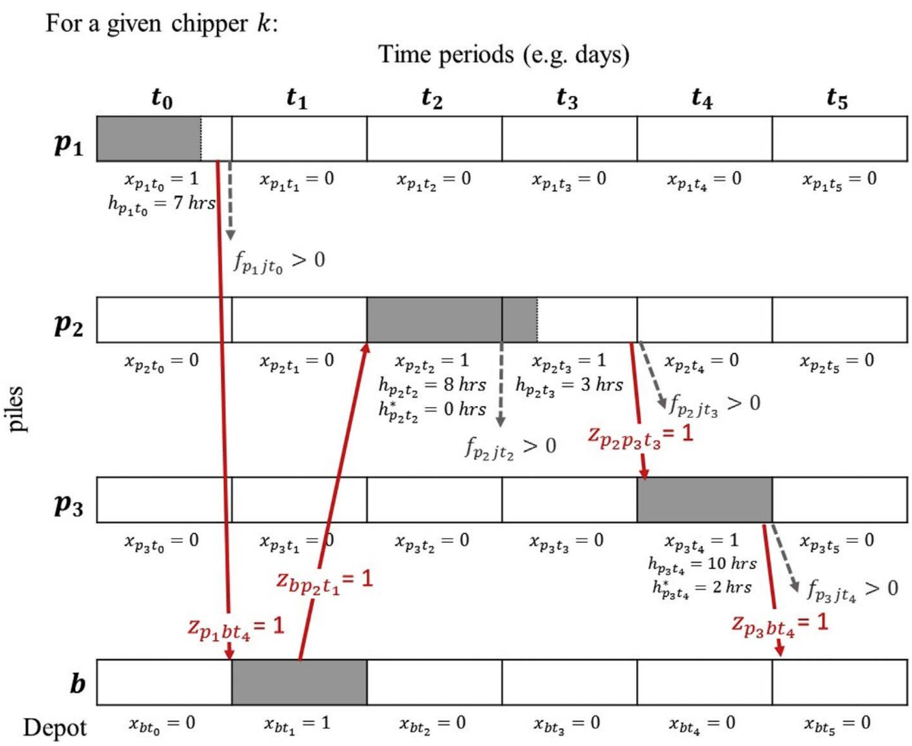


Fig. 2. Generic representation of an admissible solution for one chipper k 3 piles {p_1, p_2, p_3}, 5 time periods { t_1,..., t_5} in respect to decision variables x_{kpt}, h_{kpt}, h^*_{kpt}, $z_{kp_1p_2t}$ and f_{pjt}. For simplification purposes, the index k was omitted in the figure.

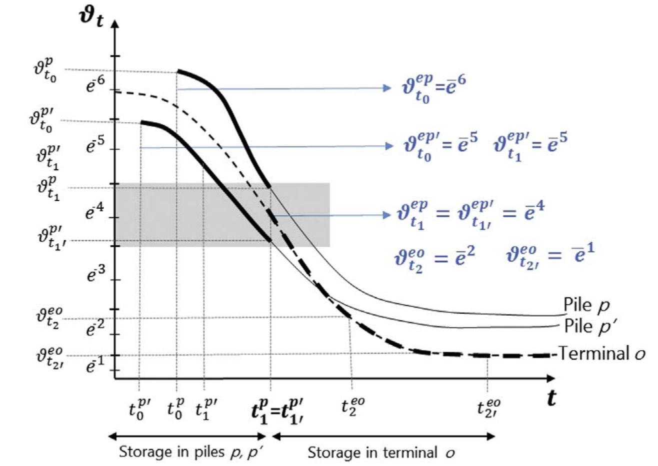


Fig. 3. Residues/wood chips drying process in the roadside pile and at the terminal Where: $\bar{e}^e$: Average moisture content of class e; ϑ_t^p, $\vartheta_t^{p'}$, ϑ_t^o: *Moisture content period t* of residues/chips located at the *roadside pile p, p' or* terminal o, respectively; t_0^p, $t_0^{p'}$: Time period since which the roadside piles p and p' are available, respectively; t_1^p, $t_1^{p'}$, $t_{1'}^{p'}$: Time period when residues/chips are transported from the piles p or p'; t_2^{eo}, $t_{2'}^{eo}$: Time periods when chips that arrived to terminal o with initial moisture content e are transported from terminal o (to plant).

3.1. Procedure for calculating moisture content variation

In this framework, a procedure is needed to compute the variation of moisture content in forest residues/chips during the time in storage (at roadside piles and/or terminals), which can be generically represented with a drying curve (Fig. 3).

Concerning storage in a roadside pile, for a given pile p the time spent in storage corresponds to $t_1^p - t_o^p$, where t_1^p is time period when residues/chips are transported from the pile to the terminal or to the plant; and t_o^p is the time period since which the pile p is available (i.e., forest harvesting operations are concluded). The initial moisture content $\vartheta_{t_0}^p$ is measured in t_o^p with an appropriate device (Assumption 1). The moisture content $\vartheta_{t_1}^p$ is estimated with the drying curve/mathematical model, which is assumed to be known (Assumption 2), as discussed in section 2.1. It is noteworthy that there can be multiple outflows from the same pile in different time periods (and for different destinations), generically represented in Fig. 3 as t_1^p, $t_1^{p'}$.

In period t_1^p, the residues/chips transportation costs (€/m3) depend on unit transportation cost τ^I (€/ton/km), distance (km) and the bulk density ρ^e (kg/bulk m^3) (Equation (4)). The latter varies with the moisture content. The price of chips at the plant (€/MWh) also depends on its net caloric value ε^e (MWh/m^3), which is calculated as a function of the moisture content (Equation (1)). Therefore, $\vartheta_{t_1}^p$ will take the average value of the corresponding class e, i.e. $\vartheta_{t_1}^{ep} = \bar{e}^e$. For implementation purposes, this can be generically represented by $\vartheta_t^p = \sum_{e\in E} \beta_{pet} \cdot \bar{e}^e$, where auxiliary binary parameters β_{pet} take the value 1 when the moisture content class $e \in E$ is applied to $p \in P$ in period $t \in T$, and 0 otherwise. These parameters are pre-processed before running the model, for all possible combinations of (p, e, t) and $\sum_{e\in E} \beta_{pet} = 1, \forall\, p, \forall\, e, \forall\, t$.

$$\tau_{ij} = \tau^I d_{ij}.\rho^e, \forall\, i \in I, \forall\, j \in J \quad (4)$$

Concerning storage in the terminal, the procedure handles batches of biomass that arrive to that terminal in the same period and belonging, in that period, to the same moisture content class, disregarding the pile of origin. This means that if the chips coming from roadside piles *p* and *p'* arrive to the same terminal in the same period and with the same moisture content class, thereafter are considered a single batch. The terminal is empty in the beginning of the planning horizon. The moisture content at the beginning of the storage time in the terminal is the same as of the departure from the pile $\vartheta_{t_1}^{ep}$, while the moisture at the time of departure from the terminal ($\vartheta_{t_2}^{eo}$) is estimated with the drying curve/mathematical model specific for that terminal, which is also assumed to be known (Assumption 2). The same approach for modelling the drying process in piles applies to terminals. The terminal may have several drying curves for distinct initial moisture content classes, and according to the season when storage began (e.g. summer or winter). Notice that there can be multiple outflows from the same batch in different time periods (and for different plants), generically represented in Fig. 3 as t_2^{eo}, t_3^{eo}.

Variables $s_{oet_1t_2}$ will deal with the stock balancing. Whenever the outflows occur, the current moisture class *e* for that period is determined, and it may be the same or lower than the one at the moment of arrival to the terminal. For example, in Fig. 3, moisture class decreased from $\bar{e}^4$ in t_1, to $\bar{e}^2$, $\bar{e}^1$ in t_2, t_3 respectively. Transportation costs and prices for the chips coming from the terminal are calculated based on current class *e*, as described before. For modelling and implementation purposes, a new auxiliary parameter is used to account for the transitions between the moisture content classes for any $(t_2 - t_1)$. The parameter $\eta_{e_1e_2t_1t_2}^o$ takes value 1, if chips arriving in terminal $o \in O$ with energy content scale $e_1 \in E$ in period $t_1 \in T^{\varnothing} = \{t | t \in T, t < t_2\}$, are expected to be in energy content scale $e_2 \in E$ in period $t_2 \in T$; 0, otherwise. The values are obtained from the drying curve/model during the model preprocessing phase, for all possible combinations of (o, e, t_1) and $\sum_{e_2\in E} \eta_{e_1e_2t_1t_2}^o = 1, \forall\, o, \forall\, e_1, \forall\, t_1$.

3.2. Mixed Integer Programming model

The formulation of the MIP model is as follows:

Sets

T	Set of planning periods, $T = \{0, \ldots, \lvert T\rvert - 1\}$
P	Set of piles of raw material at the roadside
P^b	Set of piles of raw material at the roadside, including the depot $P^b = P \cup \{b\}$
M	Set of power plants (mills)
O	Set of terminals (intermediate stockyards)
I	Set of supply points, $I = P \cup O$
J	Set of demand points, $J = M \cup O$
K	Set of chipping machines/crews
E	Set of classes of wood chips moisture content

Parameters

a_p	Availability of wood chips in pile $p \in P$ (m^3)
d_m	Demand of energy content from wood chips at plant $m \in M$ (MWh) for the entire planning period
c_m^M	Maximum throughput of wood chips at plant $m \in M$ (MWh)
c_o^O	Storage capacity in terminal $o \in O$ (m^3)
c^V	Transportation capacity of each truck (10^3 kg)
c_k^K	Unit cost of using chipper k (€)
N	Number of available trucks
r_{kp}	Productivity of chipper $k \in K$ in pile $p \in P$ (m^3/h)
y_k^m; y_k; y_k^*	Minimum, regular and max. extra-hours working time of chipper/crew $k \in K$ (h)
ω_{kp}; ω_{kp}^*	Standard and Overtime hourly chipping cost of using chipper $k \in K$ in pile $p \in P$ (€/h)
γ_o	Unit storage cost of wood chips per period at terminal $o \in O$ (€/m^3)
τ	Unit wood chips' transportation cost (€/ton/km)
χ_k	Unit transportation cost of chipper k (€/km)
d_{ij}	Distance between the point of origin i (pile or terminal) and the point of destination j (km)

ϑ_t^p	Moisture content of residues/chips in pile $p \in P$ (%) in the period t
t_0^p	Time period since which pile/depot $p \in P^b$ is available to be chipped
ρ_e	Bulk density of wood chips in energy class e (10^3 kg/m^3)
ε_e	Energy content per volume unit of wood chips in energy class e (MWh/m^3)
φ_{em}	Price paid energy content unit delivered to plant $m \in M$ (€/*MWh*)
β_{pet}	Auxiliary parameter that takes value 1 when the moisture content class $e \in E$ is applied to $p \in P$ in period $t \in T$; 0 otherwise ($\sum_{e\in E}\beta_{pet} = 1, \forall\, p, \forall\, e, \forall\, t$).
$\eta^o_{e_1e_2t_1t_2}$	Auxiliary parameter that takes value 1, if chips arriving in terminal $o \in O$ with energy content scale $e_1 \in E$ in period $t_1 \in T^\varnothing = \{t \mid t \in T,\ t < t_2\}$, are expected to be in energy content scale $e_2 \in E$ in period $t_2 \in T$; 0, otherwise ($\sum_{e_2\in E}\eta^o_{e_1e_2t_1t_2} = 1, \forall\, o,\ (e_1, t_1), t_2$)

Decision variables

x_{kpt}	1, if chipping-transportation occur in pile $p \in P^b$ in period $t \in T$, with chipper $k \in K$; 0, otherwise
$f_{ijet_1t_2}$	Amount of wood chips with energy content e transported from supply point $i \in I$ to demand point $j \in J$ in period $t_2 \in T$ that arrived in period $t_1 \in T^\varnothing = \{t \mid t \in T,\ t < t_2\}$ (m3)
$s_{oet_1t_2}$	Amount of wood chips stored at terminal $o \in O$ with energy content scale $e \in E$ in period $t_2 \in T$ that arrived in period $t_1 \in T^\varnothing = \{t \mid t \in T,\ t < t_2\}$ (m3)
h_{kpt}	Number of total hours used by machine/crew $k \in K$ in pile $p \in P$ in period $t \in T$ (h)
h^*_{kpt}	Number of overtime hours used by machine/crew $k \in K$ in pile $p \in P$ in period $t \in T$ (h)
$z_{kp_1p_2t}$	1, if chipper $k \in K$ moves from pile $p_1 \in P^b$ to $p_2 \in P^b$ at the end of period $t \in T$; 0, otherwise

Model [M1]

$$\begin{aligned}\max F = & \sum_{i\in I}\sum_{m\in M}\sum_{e\in E}\sum_{t_2\in T}\sum_{t_1\in T^\varnothing}\varphi_{em}\,\varepsilon_e\,f_{imet_1t_2} - \sum_{k\in K}\sum_{p\in P}\sum_{t\in T}C_k^K x_{kpt}\\ & - \sum_{k\in K}\sum_{p\in P}\sum_{t\in T}[\omega_{kp}\,(h_{kpt} - h^*_{kpt}) + \omega^*_{kp}\,h^*_{kpt}]\\ & - \sum_{i\in I}\sum_{j\in J}\sum_{e\in E}\sum_{t_1\in T^\varnothing}\sum_{t_2\in T}\tau\,d_{ij}\,\rho_e\,f_{ijet_1t_2}\\ & - \sum_{k\in K}\sum_{p_1\in P^b}\sum_{p_2\in P^b}\sum_{t\in T}\chi_k\,d_{p_1p_2}\,z_{kp_1p_2t}\\ & - \sum_{o\in O}\sum_{e\in E}\sum_{t_1\in T^\varnothing}\sum_{t_2\in T}\gamma_o\cdot s_{oet_1t_2}\end{aligned} \tag{5}$$

Subject to:

$$d_m \le \sum_{i\in I}\sum_{e\in E}\sum_{t_1\in T^\varnothing}\sum_{t_2\in T}\varepsilon_e\,f_{imet_1t_2} \le C_m^M \quad \forall\, m \in M \tag{6}$$

$$\sum_{j\in J}\sum_{e\in E}\sum_{t_1\in T^\varnothing}\sum_{t_2\in T} f_{pjet_1t_2} \le a_p \quad \forall\, p \in P \tag{7}$$

$$x_{kpt}\,y_k^m \le h_{kpt} \le (y_k + y_k^*)\,x_{kpt} \quad \forall\, k \in K, \forall\, p \in P, \forall\, t \in T \tag{8}$$

$$h_{kpt} - y_k \le h^*_{kpt} \le y_k^* \quad \forall\, k \in K, \forall\, p \in P, \forall\, t \in T \tag{9}$$

$$\sum_{k\in K} h_{kpt_2}\cdot r_{kp} = \sum_{j\in J}\sum_{e\in E}\sum_{t_1\in T^\varnothing} f_{pjet_1t_2} \quad \forall\, p \in P, \forall\, t_2 \in T \tag{10}$$

$$s_{oet_2t_2} = \sum_{p\in P}\sum_{t_1\in T^\varnothing} f_{poet_1t_2} \quad \forall\, o \in O, \forall\, e \in E,\ \forall\, t_2 \in T \tag{11}$$

$$s_{oet_1t_2} = s_{oet_1(t_2-1)} - \sum_{m\in M}\sum_{e'\in E}\eta_{oee't_1t_2}\,f_{ome't_1t_2} \quad \forall\, o \in O, \forall\, e \in E,\ \forall\, t_1 \in \{t \mid t \in T,\ t_1 < t_2\}, \forall\, t_2 \in T' \tag{12}$$

$$\sum_{e\in E}\sum_{t_1\in T^\varnothing} s_{oet_1t_2} \le c_o^O \quad \forall\, o \in O, \forall\, t_2 \in T \tag{13}$$

$$\sum_{i\in I}\sum_{j\in J}\sum_{e\in E}\sum_{t_1\in T^\varnothing}\rho_e\cdot f_{ijet_1t} \le c^V N \quad \forall\, t \in T \tag{14}$$

$$\sum_{p\in P^b} x_{kpt} = 1 \quad \forall\, k \in K, \forall\, t \in T \tag{15}$$

$$z_{kp_1p_2(t-1)} \ge x_{kp_1(t-1)} + x_{kp_2t} - 1 \quad \forall\, k \in K, \forall\, p_1, p_2 \in P^b\!: p_1 \ne p_2, \forall\, t \in T\backslash\backslash\{0\} \tag{16}$$

$$z_{kp_1b(t-1)} \ge x_{kp_1(t-1)} \quad \forall\, k \in K, \forall\, p_1 \in P, t = |T| \tag{17}$$

$$x_{kp_1t} + \sum_{p_2\in P^b} z_{kp_2p_1t} = x_{kp_1(t+1)} + \sum_{p_2\in P^b} z_{kp_1p_2t} \quad \forall\, k \in K, \forall\, p_1 \in P, \forall\, t \in T \tag{18}$$

$$\sum_{k\in K}\sum_{p_2\in P^b}\sum_{t\in T} z_{kp_1p_2t} \le 1 \quad \forall\, p_1 \in P \tag{19}$$

$$x_{kpt} \in \{0,1\} \quad \forall\, k \in K, \forall\, p \in P^b, \forall\, t \in T\!: t \ge t_0^p \tag{20}$$

$$0 \le f_{pjet_1t_2} \le \beta_{pet_2}\,a_p \quad \forall\, p \in P, \forall\, j \in J, \forall\, e \in E, \forall\, t_1, t_2 \in T\!: t_1 = t_0^p, t_2 \ge t_1$$

$$0 \le f_{omet_1t_2} \le \sum_{p\in P: t_1\ge t_0^p}\sum_{e_1\in E}\beta_{pe_1t_1}\,\eta_{oe_1et_1t_2}\,c_o^O \quad \forall\, o \in O, \forall\, m \in M, \forall\, e \in E, \forall\, t_1, t_2 \in T\!: t_2 > t_1$$

$$0 \le s_{oet_1t_2} \le \sum_{p\in P: t_1\ge t_0^p}\beta_{pet_1}\,c_o^O \quad \forall\, o \in O, \forall\, e \in E, \forall\, t_1, t_2 \in T\!: t_1 \le t_2$$

$$0 \le h_{kpt} \le y_k + y_k^* \quad \forall\, k \in K, \forall\, p \in P, \forall\, t \in T\!: t \ge t_0^p$$

$$0 \le h^*_{kpt} \le y_k^* \quad \forall\, k \in K, \forall\, p \in P, \forall\, t \in T\!: t \ge t_0^p$$

$$z_{kp_1p_2t} \in \{0,1\} \quad \forall\, k \in K, \forall\, p_1, p_2 \in P^b\!: p_1 \ne p_2, \forall\, t \in T : \left\{t \ge t_0^{p_1}, t+1 \ge t_0^{p_2}\right\}$$

The objective function maximizes the total profit of the biomass supplier, including the revenues from the sales of the wood chips delivered at the plants (5i) coming from the roadside piles or terminals; the daily cost of using each chipper (5ii); hourly costs of the roadside chipping operations, including the overtime work (5iii); wood chips' total transportation costs, including from the roadside to terminals or to plants and from terminals to plants (5iv); chippers' movement costs between piles (5v); and total storage costs at the terminals (5vi).

Constraints (6) define the energy content of the wood chips delivered at each power plant between the minimum demand and the maximum processing capacity. Constraints (7) state that the total flows of wood chips from piles at the roadside is at most the total availability. Constraints (8) account for the total number of chipping hours, that is upper bounded by the duration of the working shift plus the maximum allowed overtime, and lower bounded by the minimum working hours, if there is a chipper assigned to a pile. It is noteworthy that the minimum working hours in this constraint corresponds to the chippers usage cost. Below this number of hours, in principle it is more cost-efficient to conclude the work using overtime hours in the previous planning period. Complementary, constraints (9) set the number of overtime hours worked by each machinery/team, bounded by maximum allowed overtime. Constraints (10) establish the linkage between chippers' work hours and wood chips flows with origin in piles.

Constraints (11) and (12) balance the stocks per energy content at the terminals, taking into account the variation in energy content. Specifically, (11) set the amount of stock entering each terminal in each time period and moisture content class. Constraints (12) decrease the stock levels at the terminal in a given moisture content class by the amount of the outgoing flows. Which in turn are determined by the auxiliary parameter $\eta^{o}_{e_1e_2t_1t_2}$ that predicts the transition between moisture content classes, as explained in section 3.1. Constraints (13) bound the stock to the maximum capacity of the terminal. Constraints (14) bound the wood chips flows in a given period to the maximum transportation capacity available.

Constraints (15) to (18) deal with the assignment of chippers to piles and the space-time continuity of chipping operations. Specifically, Constraints (15) assure that a chipper is assigned to exactly one pile (or depot) in each time period. Constraints (16) and (17) account for the movement of the chipper between the piles (spatial continuity), where constraints (17) ensure that the chipper transportation cost for returning to the depot is considered at the end of the timeline. Constraints (18) ensure a flow conservation from t to $t+1$ in each pile p. These constraints were adapted from the balanced network flow constraints in a single machine lot sizing problem in Ref. [1]. Specifically, the two summands in the first member of the equation relate to t and cannot take value 1 simultaneously, i.e., either the chipper has already been in pile p_1 in period t ($x_{kp_1t} = 1$), or the chipper has moved to pile p_1 in the end of period t $\left(\sum_{p_2\in P^b} z_{kp_2p_1t} = 1\right)$. The two summands of the second member of the equation related to $t+1$ are also mutually exclusive and are logically constrained by t. If $x_{kp_1t} = 1$ then the chipper may continue in pile p_1 in + 1 ($x_{kp_1(t+1)} = 1$), or move to another pile p_2 at the end of period t $\left(\sum_{p_2\in P^b} z_{kp_1p_2t} = 1\right)$. If $\sum_{p_2\in P^b} z_{kp_2p_1t} = 1$, then the chipper will necessarily stay in pile p_1 in period $t+1$, as imposed by constraints (16). Constraints (19) assure that there is at the most one movement from each pile p, along the entire time horizon.

Constraints (20) set the domain of the decision variables. These constraints take advantage of the previously defined parameters in order to eliminate non-admissible decision variables. For example, wood chips' flows from piles (or terminals) to power plants in a certain moisture content class are unfeasible if wood chips never reach those moisture content values. It is noteworthy that, although variables $z_{kp_1p_2t}$ take binary values, they may be linearized to improve the model performance.

3.3. Model variations

A simplification of the model is possible if the variation of the moisture content of the biomass at the terminals is not explicitly dealt in the model as a function of the storage age. In fact, there may be situations where the terminal is managed in a way that assures that the moisture content of the chips coming out of the terminal is fixed and previously determined in the beginning of the planning process. This may happen for example in large terminals, or when technical drying is used. In this case, the moisture content at the terminals in each time period is a parameter of the model, and is independent from the characteristics of the incoming flows that are set by the model.

Hence, the moisture content in each time period is a parameter varying with the terminal, which can be defined by the user in the beginning of the planning process. Model [M2] is significantly reduced in this aggregated approach since the set of classes of wood chips moisture content does not need to be considered and it is not necessary to keep track of the storage age of the wood chips batches. Thus, variables $f_{ijet_1t_2}$ are replaced by f_{ijt}, representing the amount of wood chips transported from origin $i \in I$ to destination $j \in J$ in period $t \in T$ (m^3). While variables $s_{oet_1t_2}$ are replaced by s_{ot}, amount of wood chips stored at terminal $o \in O$ in the end of period $t \in T$. Similarly, parameters ε^e, ρ^e, φ_{em}, are replaced by ε_{it} (energy content per volume unit of wood chips at origin $i \in I$ in period $t \in T$ (MWh/m3)), ρ_{it} (bulk density wood chips at origin $i \in I$ in period $t \in T$ (kg/m3)), and φ_{imt} (Price paid per volume unit of wood chips with origin in $i \in I$, delivered to power plant $m \in M$ in period $t \in T$ (€/m^3)). Consequently, in model [M2] the objective function is simplified (5b) and constraints (6), (7), (11) to (14) are replaced by (6b), (7b), (11b) to (14b). New constraints (21) are needed to assure that the wood chips remain in the terminal for at least one period.

Model [M2]:

$$\max F = \sum_{i\in I}\sum_{m\in M}\sum_{t\in T}\varphi_{imt}\cdot f_{imt} - \sum_{k\in K}\sum_{p\in P}\sum_{t\in T}\left[\omega_{kp}\cdot(h_{kpt} - h^*_{kpt}) + \omega^*_{kp}\cdot h^*_{kpt}\right] - \sum_{k\in K}\sum_{p\in P}\sum_{t\in T} C^K_k x_{kpt} - \sum_{j\in I}\sum_{j\in J}\sum_{t\in T}\tau_{ijt}\cdot f_{ijt} - \sum_{k\in K}\sum_{p_1\in P^b}\sum_{p_2\in P^b}\sum_{t\in T}\chi_k\, d_{p_1p_2}\, z_{kp_1p_2t} - \sum_{o\in O}\sum_{t\in T}\gamma_o\cdot s_{ot} \quad (5b)$$

Subject to: constraints (8), (9), (15), (16), (17), (18), (19),(20) and

$$d_m \le \sum_{i\in I}\sum_{t\in T}\varepsilon_{it} f_{imt} \le C^M_m \quad \forall\, m \in M \quad (6b)$$

$$\sum_{j\in J}\sum_{t\in T} f_{pjt} \le a_p \quad \forall\, p \in P \quad (7b)$$

$$\sum_{k\in K} h_{kpt_2}\cdot r_{kp} = \sum_{j\in J}\sum_{t_1\in T^{\varnothing}} f_{pjt_1t_2} \quad \forall\, p \in P, \forall\, t_2 \in T \quad (10b)$$

$$\sum_{p\in P} f_{pot} = s_{ot} \quad \forall\, o \in O, \forall\, t \in T\colon t = 0 \quad (11b)$$

$$s_{o(t-1)} + \sum_{p\in P} f_{pot} - \sum_{m\in M} f_{omt} = s_{ot} \quad \forall\, o \in O, \forall\, t \in T\backslash\backslash\{0\} \quad (12b)$$

$$s_{ot} \le C^O_o \quad \forall\, o \in O, \forall\, t \in T \quad (13b)$$

$$\sum_{i\in I}\sum_{j\in J}\rho_{it}\cdot f_{ijt} \le C^V \quad \forall\, t \in T \quad (14b)$$

$$\sum_{m\in M} f_{omt} \le s_{o(t-1)} \quad \forall\, o \in O, \forall\, t \in T\backslash\backslash\{0\} \quad (21)$$

Another possible model variation is a simplification of the movements of the chippers between piles (constraints 16) to improve model performance in case of instances of significant size. In fact, constraints (16) are computationally expensive due to the exponentially increasing number of $z_{kp_1p_2t}$ decision variables when the number of piles increases. The proposed simplification approach consists in outlining a set of Q pre-defined geographical and non-overlapping neighborhoods for each pile p - $\psi_p = \psi_p^0 \cup \psi_p^1 \cup ...\cup\psi_p^{Q-1}$ - given some distance radius criteria and re-set constraints (16) to (18) accordingly. For example, considering a certain pile p, where $Q = 4$: the first neighborhood $\left(\psi_p^0\right)$ could be composed by all piles within 20min distance of pile p, the second neighborhood $\left(\psi_p^1\right)$ by piles within 40min distance (not including the ones already contained in ψ_p^0) and ψ_p^2 by all remaining piles. For each one of these neighborhoods we compute the average of the distances between pile p and the locations in the neighborhood, which is then multiplied by the unit chippers' transportation cost to be incorporated in the model's objective function. Additionally, and due to the special characteristics of this location, an additional neighborhood $\left(\psi_p^3\right)$ containing only the depot is also set, so that this location can still be distinguished within the model.

Consequently, in the new model formulation [M3], decision variables $z_{kp_1p_2t}$ would be replaced by variables z_{kpqt}, taking value 1, if chipper $k \in K$ moves from pile $p \in P$ to another location contained in neighborhood ψ_p^q at the end of period $t \in T$; 0, otherwise. Constraints (16)–(18) would change to constraints (16b)-(18b) as described below.

Model [M3]:

Objective function (5) or (5b)

Subjected to: constraints (6) to (15), (20) and

$$z_{kp_1q(t-1)} \geq x_{kp_1(t-1)} + \sum_{p_2 \in \psi_p^q} x_{kp_2t} - 1 \;\; \forall\, k \in K, \forall\, p_1 \in P^b, \forall\, q, \;\; \forall\, t \in T \backslash\backslash \{0\} \tag{16b}$$

$$z_{kp_1q(t-1)} \geq x_{kp_1(t-1)} \;\; \forall\, k \in K, \forall\, p_1 \in P, \, q = Q-1, \;\; t = |T| \tag{17b}$$

$$x_{kp_1t} \leq x_{kp_1(t+1)} + \sum_{q \in Q} z_{kp_1qt} \;\; \forall\, k \in K, \forall\, p_1 \in P, \forall\, t \in T \tag{18b}$$

$$\sum_{k \in K} \sum_{q \in Q} \sum_{t \in T} z_{kp_1qt} \leq 1 \;\; \forall\, p_1 \in P \tag{19b}$$

4. Computational experiments

The proposed model [M3] was applied to a case study inspired in a wood chips supplier company operating in Southern Finland. The model was implemented in the Gurobi 7.5.1 solver and was run in a 2.60 GHz CPU with capacity for 32 simultaneous processing threads. Model M3 was tested with three different planning horizons were tested: 1 month (40 half-day planning periods), 1.5 months (60 periods) and 2 months (80 periods). Then, the results for 1 month were analyzed in detail. An additional computational experience was conducted to compare the proposed planning approach under variable wood chips moisture content with the baseline situation that relies on empirical estimates for a fixed and known moisture content in the end of an obliged storage age.

4.1. Case study

The company manages their own chipping and transport operations, based on biomass supply contracts with the power plants. The company acquires the piles of forest residues that are byproducts of harvesting operations. The case under study encompasses 84 piles of forest residues of spruce, geographically distributed and ready to be chipped. Because a significant number of piles have very low availabilities and their distances to other piles are residual in some cases, a clustering procedure was implemented in order to reduce the problem size. After this procedure, a total number of 55 clusters of piles were considered, hereafter called macro-piles. The macro-pile location is generically represented by the centroid of the circular cluster, computed with the GIS software based on the location of its piles; the movements of the chippers between the piles of the same cluster are neglected. The problem also contains 4 terminals and 12 power plants. The terminals are assumed empty in the beginning of the planning period. There are also 9 chippers available to do the work. Planning periods are halves of a day - 3.5 h, assumed to be the length of the regular chipping shift. The parameters of the model [M3] are summarized in Table 1.

This case considers 5 moisture content classes (class $\boldsymbol{e}_1 \in [15\%, 20\%[$ to $\boldsymbol{e}_5 \in [50\%, 60\%[$). The bulk density for the average value of the class e (ρ^e) (kg/bulk m^3) is given by reference values for the most common wood fuels (e.g. Ref. [15]). Similarly, the wood chips net caloric value for each class ε^e (KWh/m^3) is computed with Equation (1) in respect to $\bar{e}^e$. The moisture content variation was obtained with the logistics function (equation (2)) adjusted for the data set of [21] as described in section 2.1. For each week of the planning period, moisture content was calculated, framed into its corresponding moisture content class and parameters β_{pet} and $\eta^o_{e_1e_2t_1t_2}$ were set accordingly.

The other parameters of the model were inspired in a former biomass plan done by the company, including the location and availability of the piles; the location, demand and throughput of the plants, the location and capacity of the terminals; number and type of chippers, working hours and transportation capacity. Chipping and storage costs were inspired in the company business and perhaps below the range of the values found in the literature (e.g. Ref. [15]). The transportation costs were computed with Equation (4), considering a unit transportation cost of 0,08 €/m3/km. The distances between locations were computed by resorting to the national road dataset of Finland (http://www.liikennevirasto.fi/avoindata/digiroad#.WPi-W1OGPfC). Prices of wood chips vary according to its moisture content and the power plant they are delivered to. These prices were estimated considering a fixed price of 13€/MWh [9]. Only 7 of the 12 power plants accept wood chips regardless of their moisture content. Note that, although in this particular case the price is fixed, there is still an incentive for the model to opt for wood chips delivery of lower moisture content, as the price is fixed by energy content unit.

In sum, the entire instance of 80 periods exhibits a total availability of approximately 19,050 m^3 of spruce wood chips in 55 macro-piles and the total demand in the 9 power plants is 26,951 MWh (corresponding to 16,844 m^3 of chips in the higher moisture class accepted by the power plants). Total available storage capacity is 52,000 m^3. The chipping capacity ranges between 119 m^3 and 240 m^3 per planning period and maximum transportation capacity per period is 600ton.

Table 1
Values of the parameters of the model in the case study.

Sets/Parameter	Value
T: Planning periods	Varies between 40 and 80 (halves of a day)
E: Classes of moisture content	$\boldsymbol{e}_1 \in [15{,}20[$;$\boldsymbol{e}_2 \in [20{,}30[$;$\boldsymbol{e}_3 \in [30{,}40[$; $\boldsymbol{e}_4 \in [40{,}50[$; $\boldsymbol{e}_5 \in [50{,}60[$
ρ^e: Bulk density for each class e	$\rho^1 = 354$, $\rho^2 = 424$, $\rho^3 = 483$, $\rho^4 = 572$, $\rho^5 = 632$ kg/m^3
ε^e: Wood chips net caloric value	$\varepsilon^1 = 1.92$, $\varepsilon^2 = 1.87$, $\varepsilon^3 = 1.8$, $\rho^4 = 1.73$, $\varepsilon^5 = 1.47$ MWh/m^3
P: Macro-Piles	**55**
a_p: Chips availability	Varies between 10 and 2500 bulk m^3
ϑ_0^p: Initial moisture content	Varies between 30% and 50%
t_0^p: Time since when pile is available	Varies between period 1 and 40
M: Power plants (or mills)	**12**
d_m: Demand	Varies between 512 and 6144 MWh for the total planning horizon
c_m^M: Throughput	5% more than the demand in each plant
φ_{em}: Price of wood chips	13€/MWh
T: Terminals	**4**
c_o^O: Capacity	Varies between 4000 and 40000 m^3
γ_o: Unit storage cost	Varies between 0,10 and 1,00€/m^3
K: Chippers	**3**
r_{kp}: productivity	Varies according to chipper and pile, between 34 and 48 bulk m^3/hour
y_k^m; y_k; y_k^*: Min, regular, max working hours	0; 3,5; 0,5 h, equal for all chippers
ω_{kp}; ω_{kp}^*: Regular, overtime hourly chipping cost	300; 450 €/hour, equal to all piles
χ_k: Unit chipper transportation cost	3,0 €/km, equal for all chippers
C_k^K:Chipper usage cost	700€/period, equal for all chippers
Wood fuel Transportation	
c^V: Truck transportation capacity	30 ton
N: Number of available trucks	20
τ: Unit transportation cost	0,08 €/ton/km

Table 2
Computational results for model [M3], for 40 to 80 planning periods.

Instance Size				Model size			Computational time		
T	I	a m^3	D MWh (m^3)	# bin.	# cont.	# const.	OF	Runtime (s)	Gap %
40	17	16819	13475 (7018–11229)	2529	49462	21870	89465	9978	0.01
60	45	18098	20213 (10527–16705)	6705	128451	63486	129962	172800	0.13
80	55	19050	26951 (14036–22274)	12699	242020	122501	166996	172800	0.91

T: N. time periods, I: N. macro-piles available up to the end of T; a: total availability of wood chips up to T (m^3); d: total demand up to T (in MWh and the range of corresponding m^3); # bin.: number of binary variables; # cont.: number of continuous variables; # const.: Number of constraints in the model; FO: value of the objective function (€); Runtime until optimality proven (sec.).

4.2. Comparison of the model performance and results for 40, 60 and 80 time periods

Considering the three different planning horizons, the total demanded MWh increase proportionally to the number of planning periods. The wood chips availability also increases because at the beginning of the planning horizon a minority of the piles are available to be chipped, therefore, the longer the time horizon, the higher number of piles available (see Table 2).

The instance with 40 periods (17 macro-piles, demand of 13,475 MWh) results in a MIP problem with 2529 binary variables, 49,462 continuous variables and 21,870 constraints, which is solved to optimality in approximately 2 h and 45 min. The total profit of the biomass supplier is 89,465 €. The profit gained at the end of 60 periods (45 macro-piles, demand of 20,213 MWh) increases to 129,962 €. In this case, the M3 model has 38% more decision variables, 190% more constraints and takes 48 h to solve the problem to a gap of 0.13% (Table 3). For the instance with 80 periods (55 macro-piles, demand of 26,951 MWh), the model size increases to 12,699 binary variables, 242,020 continuous variables and 187,440 constraints, which reaches the objective function value of 166,996 € with a gap of 0.91% after 48 h. The analysis of the performance of the model suggests that this approach is adequate to solve problems to up to 40 periods (1 month) in a common computer, thus requiring more sophisticated solution approaches for larger problem instances.

In respect to the comparison of the model results, the number of chippers used along the planning horizon varies between 2 (40 periods) and 7 (80 periods) and the average number of trucks is kept steady around 5 trucks per period. This happens because the chipper usage cost is considerably high and the optimal solution will favor using a chipper's capacity to its maximum instead of resorting to additional chippers. In these instances, terminals have excess capacity, so one terminal is sufficient in all test cases to accommodate all wood chips transshipment flows from macro-piles.

Table 3
Results for model M3, for 40, 60 and 80 planning periods.

T	I'	a' m^3	d' MWh (m^3)	k	h hours	s	G periods	n
40	9	7505	14150 (7505)	2	176	1	6.7	94.7
60	13	11273	21224 (11273)	3	273	1	5.8	91.1
80	17	15050	28299 (15050)	7	367	1	6.2	87.8

T: N. time periods, I': N. macro-piles used; a': total availability of wood chips effectively used (m^3); d': total demand fulfilled (MWh and m^3); k: N. chippers; used; h: total number of working hours, including overtime work; s: N. of terminals used; g: average storage age (periods); n: average transportation capacity used per period (ton).

As time periods increase, it is also observable that the average storage age in piles and terminals tends to decrease. This fact is hardly justified by the model behavior and seems to be related with the characteristics of the instance under study. In fact, many piles become available (after harvesting) closer to the end on the planning horizon. Considering the 80 period's instance, the average period when piles become available is 43.4. Furthermore, the difference between piles' availability and power plants' demand decreases significantly and it forces the wood chips to be delivered to the plants sooner, thus spending less time in storage.

4.3. Results of the test case of 40 periods

In the 40 periods instance, the total profit from wood chips sales sums up 89,465€, while total costs sum 98,799€, corresponding to 8297€ of wood chips transportation costs, 55,533€ of chipping costs, 31,500€ of chipper usage costs, 3127€ of chipper transportation costs and 342€ of storage costs. Therefore, 12% of the costs relate to transportation and 88% of the costs to chipping.

Chipping operations: require 2 chippers and a total of 176 chipping hours to produce 7504 m^3 of wood chips in 9 macro-piles (corresponding to 18 piles). The schedule of chipping and transportation roadside in macro-piles is presented in Fig. 4a). For example, in macro-pile 19 chipping starts in period 14 with chipper 3 and extends up to period 19, resulting in 1008 m^3 of wood chips, of which 90 m^3 are transported directly to plant 11 with moisture content e_3 [30–40%], and 918 m^3 are transported to plant 12, arriving there also with moisture content class e_3. All the selected macro-piles require more than one period to be fully chipped. This schedule confirms that temporal continuity constraints are fully satisfied as chipping extends over consecutive periods.

A complementary view of the chipper's schedules shows the optimum sequence of macro-piles for the selected chippers along the planning horizon (Fig. 4b). For example, chipper 6 starts working in period 23 in macro-pile 42, moves to macro-pile 6 in the end of period 26, then to macro-pile 3 in the end of period 30 and finally to macro-pile 9 in the end of period 35. The total chipping hours are 68.7, including 5.7 h of overtime work. The total chipping cost associated to the daily work of chipper 6 is 21,462 €. The schedule shows that spatial continuity constraints are fully satisfied as the chipper moves to neighboring clusters after all the available amount of forest residues in the pile is chipped and transported. It is noteworthy that chipping operations only start after period 14. This is due to the fact that, as expected, the optimal solution fulfills exactly all the demand at the plants and delays chipping as much as possible, to take advantage of the decrease of moisture content while in storage at the piles or terminals. Despite that fact, the chipping capacity in the last planning periods remains sub-used due to the high daily chipper utilization costs.

The trade-off between the chipper utilization costs (700€/period) and the cost of regular and overtime work (300 €/hour and 450 €/hour

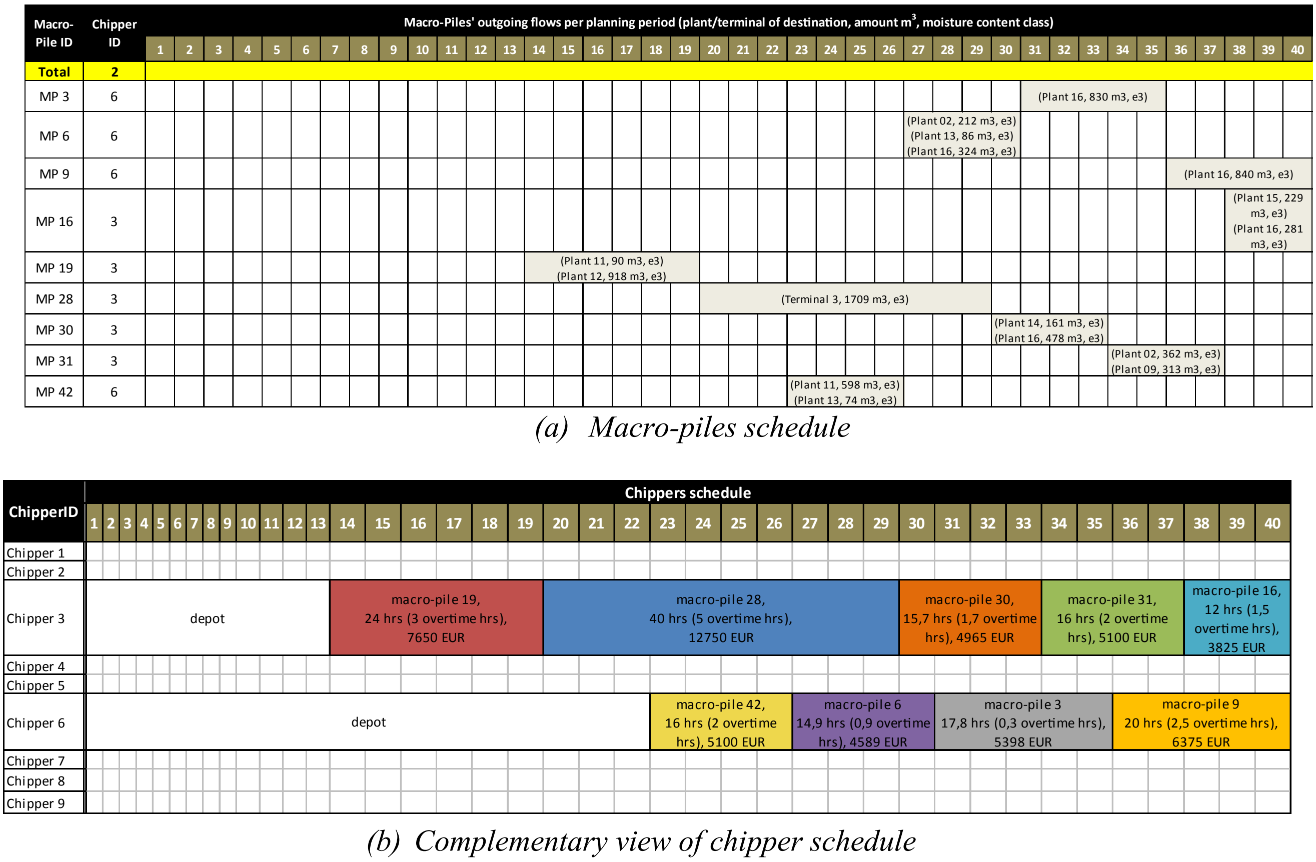
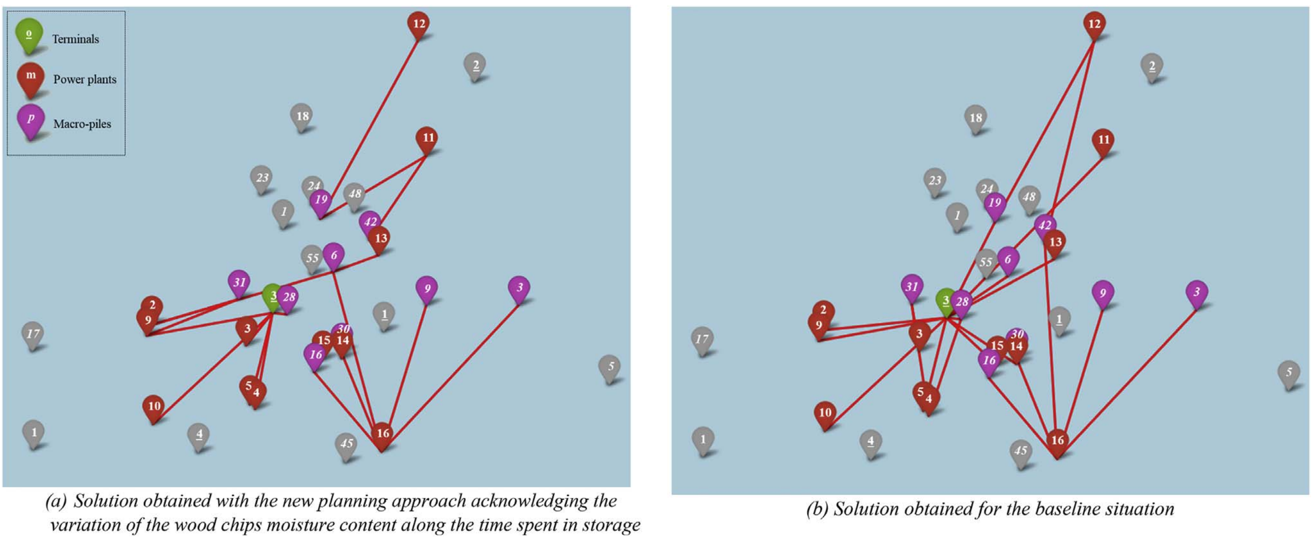


Fig. 4. Schedule chipping and transportation operations over the 40 periods in the 7 selected macro-piles with the 2 selected chippers.

respectively) is shown in the model results. For example, chipper 6 remains in macro-pile 9 for 5 periods and uses a total of 17.5 h of regular work and 2.5 h of overtime, corresponding to a cost of 5250 € for regular work and 1125 € for overtime work, which is still lower than the additional cost of 1525 € for chipping the remaining amount in another period (700 € for an additional period, plus 600 € for regular work and 225 € for overtime work).

Transportation of wood chips/forest residues: the average transportation capacity used per period is 94.7 ton, corresponding to 5 trucks. Transportation starts in period 14, which is also when chipping starts, thus confirming the consistency of the model in respect to the synchronization of these roadside operations. From period 14 up to period 21, the transportation flows per period are 72 ton approximately, corresponding to 4 trucks. Then, here is significant increase of the transportation flows until reaching a maximum of 204 ton (11 trucks) in periods 23–28. From then up to the end of the timeline the transportation flows decrease and stabilize to a range corresponding to the use of 8 trucks. The total transportation network is presented in Fig. 5a).

Storage: only terminal 3 is used, because it is the one that is closest



(a) Solution obtained with the new planning approach acknowledging the variation of the wood chips moisture content along the time spent in storage

(b) Solution obtained for the baseline situation

Fig. 5. Transportation network between macro-piles, terminals and power plants for all the periods.

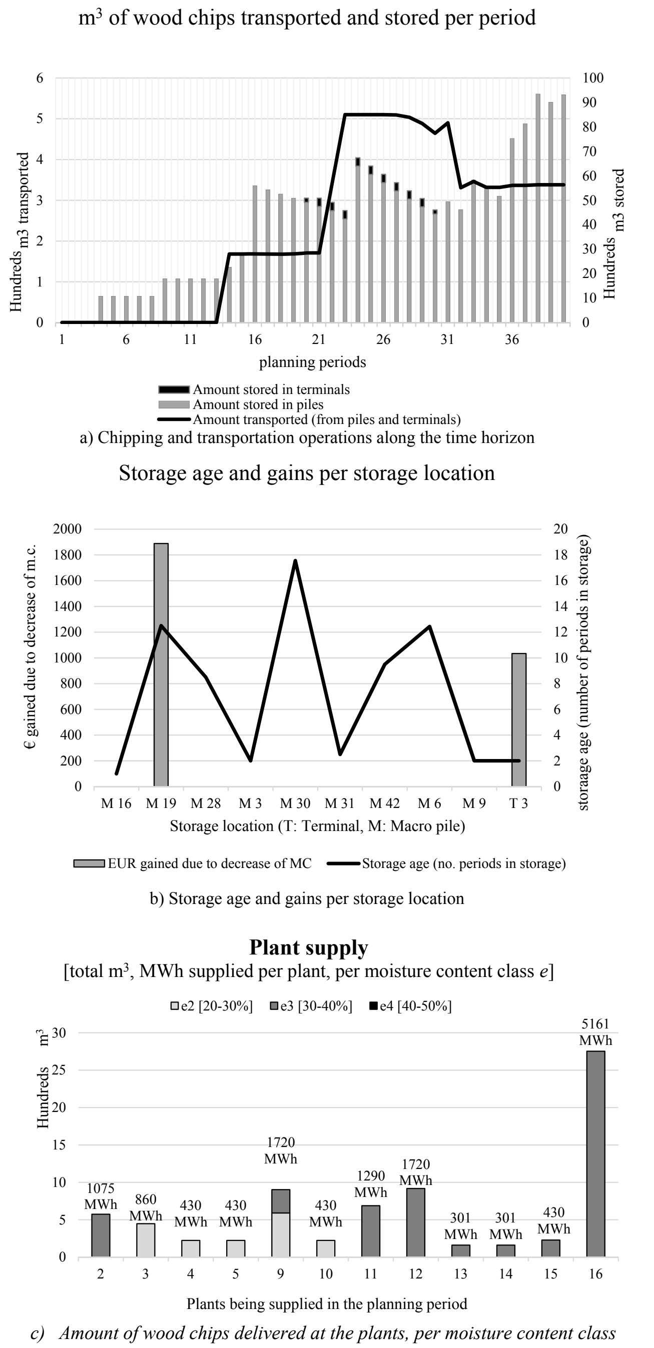


Fig. 6. Chipping, transportation, storage and plant supply for the case of 40 periods.

to the macro-piles that are used. Furthermore, this terminal is the one that has the lowest utilization costs. Still the level of its utilization is very low. The maximum storage capacity used is 342 m^3 between periods 20 and 28, significantly below the terminal capacity (4000 m^3). There are only 10 incoming flows for terminal 3 (between periods 20 and 29 from macro-pile 28, with moisture content e3). The average storage age is 2 periods, which is most likely a consequence of the parameters of the drying curve used in this case. For the forest residues stored in piles, the average storage age is 6.7 periods (Table 3).

The distribution of the transportation and storage amounts along the time horizon is presented in Fig. 6a). The piles are the main stocking location. There is an average of 4161 m^3 stocked in the piles per period. The maximum of the stock occurs in period 38 when most of the piles used are already available.

In respect to the profit increase due to the loss of moisture content during the time in storage, the 1709 m^3 stored in terminal 3 correspond to an increase of profit of 1034€. The gain due to storage in the roadside piles is 1888€, corresponding to the decrease of moisture content from e5 to e3 in macro pile 19 (Fig. 6b)). The total gain sums up 2923€, about 3.3% of the total profit. It is noteworthy that this value may be under-estimated since the decrease of moisture content for the remaining macro piles did not correspond to a reduction of the moisture content class, consequently the gain is not quantified. This is a drawback of the discretization approach with the class length corresponding to 10% variation. When the initial moisture content is closer to the upper limit of the moisture class, the short storage age may not be enough to decrease to a lower class, while if the moisture content is closer to the lower limit of the class it will likelier decrease.

Demand fulfillment: As expected in this demand-driven problem, the amount supplied is equal to the maximum amount demanded by the plant, summing 14,150 MWh (Fig. 6c)). The majority of the plants was exclusively supplied with wood chips with moisture content e3 [30–40%], except for plants 3 to 5 and 9 to 10, which receive a total amount of 1709 m^3 of wood chips with lower moisture content from terminal 3This can be justified due to the fact that these plants are the closest ones terminal 3, therefore benefitting from the fact that the wood chips have a longer drying age and were subjected to most favorable drying conditions (i.e. better drying curve) in that terminal than in other terminals or roadside piles.

4.4. *Comparison of the proposed planning approach under variable wood chips moisture with the baseline situation*

The baseline situation for supply planning corresponds to relying exclusively on empirical estimates for a fixed and known moisture content in the end of a fixed storage age that may vary according to each storage location. This is often the case when portable devices are not available or not frequently used to monitor the moisture content variation along the time spent in storage. In opposite, model [M3] used in this study explicitly handles the variation of moisture content along the variable time in storage, thus representing a change in the company current business practices that can help to reduce operational costs.

In this study, we assumed a baseline corresponding to an initial moisture content in the wood piles (after harvesting) of 48%, unchangeable regardless the time remaining in storage; and of 30% after 10 periods spent drying in a given terminal. The steps to simulate the baseline include: 1) Adapt the data set to consider the empirical estimates of moisture content of 48% in the wood piles, unchangeable regardless the time remaining in storage; and of 30% after 10 periods spent drying in a given terminal; 2) Use [M2] to compute values of the decision variables x_{kpt}, h_{kpt}, h^*_{kpt}, $z_{kp_1p_2t}$, f_{ijt}, s_{ot} under those assumptions; 3) Compute the $f_{ijet_1t_2}$ equivalent to f_{ijt} and the $s_{oet_1t_2}$ equivalent to s_{ot} and then inject the values of the decision variables x_{kpt}, h_{kpt}, h^*_{kpt}, $z_{kp_1p_2t}$, $f_{ijet_1t_2}$, $s_{oet_1t_2}$ in [M3] to compute the value of the objective function.

The value of the objective function for the BAU obtained with this procedure is 70,475€. This is 21.2% less than the total profit obtained with our proposed approach that explicitly handles the variation of moisture content along the variable time in storage (89,465€).

In fact, our approach shows that it is more cost-effective to leave the wood chips to dry in the wood piles at the roadside. The flows to terminals are significantly reduced and the chips remain there 2 periods, in average, instead of 10 as considered in the BAU. Consequently, wood chips transportation costs are 15,138€, 82% higher than with the proposed approach, not only because of the increase in chips' moisture content in piles, but also sue to the additional transport of wood chips to terminals. The average transportation capacity used per period increases from 156 ton (8 trucks) to 347 ton (18 trucks). Similarly, storage costs in the BAU are 3255€, more than 850% higher than with the proposed approach. This is because the BAU forces the material that goes to terminal 3 to remain there for at least 10 periods. The total transportation network for the 40 periods is presented in Fig. 5b). By comparing both maps we can conclude that chippers perform chipping in the same piles, and the optimal solution in the BAU favors visits to terminal 3, while in the proposed approach the terminal is only used if chips' delivery is made in a nearby location.

5. Concluding remarks

This paper presents a novel mathematical programming model for tactical biomass supply planning problem, in case of synchronized chipping and transportation at the roadside ("hot systems"), and explicitly considering the variation in chips energy content (or moisture content) over time in storage. It builds on previous research from Refs. [19] and [14]. 2 model variations were discussed. The first, does not account for moisture content classes, as it assumes that moisture content of the chips coming out from storage is a user-defined parameter, known at the beginning of the planning process. The second, is an simplified way to model the movement of the chipper between piles that considers geographic neighborhoods for each pile that at defined beforehand. This approach reduces the number of constraints, therefore improving the model performance for larger problem instances. The latter modelling approach is successfully used to solve to optimality within 3 hoursrs, problem instances with 17 macro-piles, 40 periods (half days), 4 terminals (for intermediate storage) and 12 power plants. 6 moisture content classes were considered, ranging from 30% to 55%. 9 chippers (with heterogeneous productivity) and 20 trucks (with homogenous capacity) were available. The results were presented in the form of a chipping-transportation schedule for each macro-pile/chipper and a flow map. Results suggest that a 20% improvement in the supplier profit can be obtained with the proposed approach that explicitly handles the variation of moisture content along the variable time in storage, when compared with a baseline situation that relies on empirical estimates for a fixed and known moisture content in the end of an obliged storage age.

Future work will seek alternative modelling approaches to solve larger instances that characterize real-life planning situations (e.g. 145 piles and up to 350 periods). For this purpose, a novel formulation will be developed, inspired in the generalized lotsizing and scheduling problem [13]. Alternative heuristic procedures may also be considered. Robust optimization or stochastic programming may be tested to address relevant uncertainty sources, both at the supply level (e.g. uncertainty in the moisture content values in each pile) and at the demand level (e.g. MWh demanded by each plant). Future work may also seek for optimizing the daily routing and scheduling for each chipper and truck (at hourly level), taking into account the synchronization constraints (e.g. Ref. [7]).

Acknowledgments

This research has received funding from the European Union Seventh Framework Programme (FP7/2007–2013) under grant agreement number [604286]. This work is also financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project POCI-01-0145-FEDER-016733 (Easyflow). Further funding was obtained from the Project "NORTE-01-0145-FEDER-000020 (TEC4Growth - Pervasive Intelligence, Enhancers and Proofs of Concept with Industrial Impact)", financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF)European Regional Development Fund.

References

[1] Bernardo Almada-Lobo, Diego Klabjan, Maria Antónia Carravilla, José F. Oliveira, Single machine multi-product capacitated lot sizing with sequence-dependent setups, Int. J. Prod. Res. 45 (20) (2007) 4873–4894 2007 https://doi.org/10.1080/00207540601094465Oliveira.
[2] P. Amorim, A.M. Costa, B. Almada-Lobo, Influence of consumer purchasing behaviour on the production planning of perishable food, OR Spectr. 36 (3) (2014) 669–692 https://doi.org/10.1007/s00291-013-0324-9.
[3] Antti Asikainen, Simulation of stump crushing and truck transport of chips, Scand. J. For. Res. 25 (3) (2010) 245–250 https://doi.org/10.1080/02827581.2010.488656.
[4] N. Atashbar, N. Zandi, Labadie, C. Prins, Modeling and optimization of biomass supply chains: a review and a critical look, IFAC-PapersOnLine, 8th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2016Troyes, France, 28—30 June 2016, vol. 49, 2016, pp. 604–615 12 https://doi.org/10.1016/j.ifacol.2016.07.742.
[5] M.D. Casal, M.V. Gil, C. Pevida, F. Rubiera, J.J. Pis, Influence of storage time on the quality and combustion behaviour of pine woodchips, Energy 35 (7) (2010) 3066–3071 https://doi.org/10.1016/j.energy.2010.03.048.
[6] Annelies De Meyer, Dirk Cattrysse, Jussi Rasinmäki, Jos Van Orshoven, Methods to optimise the design and management of biomass-for-bioenergy supply chains: a review, Renew. Sustain. Energy Rev. 31 (2014) 657–670 https://doi.org/10.1016/j.rser.2013.12.036.
[7] Michael Drexl, Applications of the vehicle routing problem with trailers and transshipments, Eur. J. Operational Res. 227 (2) (2013) 275–283 https://doi.org/10.1016/j.ejor.2012.12.015.
[8] A. Dunnett, C. Adjiman, N. Shah, Biomass to heat supply chains: applications of process optimization, Process Saf. Environ. Prot. 85 (5) (2007) 419–429 https://doi.org/10.1205/psep07022.
[9] Energy Prices, Official Statistics of Finland, (2016) http://www.stat.fi/til/ehi/index_en.html.
[10] Gernot Erber, Christian Kanzian, Karl Stampfer, Predicting moisture content in a pine logwood pile for energy purposes, Silva Fenn. 46 (4) (2012), https://doi.org/10.14214/sf.910.
[11] Eriksson, Improving the Efficiency of Forest Fuel Supply Chains, Doctoral Thesis (2016) Uppsala.
[12] A. Eriksson, L. Eliasson, R. Jirjis, Simulation and Modeling of Wood Chip Container Logistics at Forest Landings, (2014) BennetOY http://www.bioenergyevents.fi/default.asp?sivuID=28594&component=/_ACC/_Components/ATLANTIS-digiStore/default_embed.asp&ID=3827&settings=.
[13] Bernhard Fleischmann, Herbert Meyr, The general lotsizing and scheduling problem, Operations-Research-Spektrum 19 (1) (1997) 11–21 https://doi.org/10.1007/BF01539800.
[14] P. Flisberg, M. Frisk, M. Rönnqvist, FuelOpt: a decision support system for forest fuel logistics, J. Operational Res. Soc. 63 (11) (2012) 1600–1612 https://doi.org/10.1057/jors.2011.157.
[15] Valter Francescato, Eliseo Antonini, Wood Fuels Handbook, AIEL, 2008, www.biomasstradecentres.eu.
[16] Shuva Gautam, Luc LeBel, Marc-André Carle, Supply chain model to assess the feasibility of incorporating a terminal between forests and biorefineries, Appl. Energy 198 (Supplement C) (2017) 377–384 https://doi.org/10.1016/j.apenergy.2017.01.021.
[17] Stefan Gold, Stefan Seuring, Supply chain and logistics issues of bio-energy production, J. Clean. Prod. 19 (1) (2011) 32–42 https://doi.org/10.1016/j.jclepro.2010.08.009.
[18] Manfred Gronalt, Peter Rauch, Designing a regional forest fuel supply network, Biomass Bioenergy 31 (6) (2007) 393–402 https://doi.org/10.1016/j.biombioe.2007.01.007.
[19] Helene Gunnarsson, Mikael Rönnqvist, Jan T. Lundgren, Supply chain modelling of forest fuel, Eur. J. Operational Res. 158 (1) (2004) 103–123 https://doi.org/10.1016/S0377-2217(03)00354-0.
[20] Hofmann, Nicolas, Theresa Mendel, Fabian Schulmeyer, Daniel Kuptz, Herbert Borchert, and Hans Hartmann. n.d. "Drying Effects and Dry Matter Losses during Seasonal Storage of Spruce Wood Chips under Practical Conditions." Biomass and Bioenergy. Accessed April 10, 2017. https://doi.org/10.1016/j.biombioe.2017.03.022.
[21] A. Holdrich, H. Hartmann, Rationelle Scheitholzbereitstellungsverfahren, (2006) Technologie-und Forderzentrum (TFZ) http://www.tfz.bayern.de/mam/cms08/festbrennstoffe/dateien/11_bericht.pdf.
[22] Fernando Jaramillo, Murat Erkoc, Minimizing total weighted tardiness and overtime costs for single machine preemptive scheduling, Comput. Industrial Eng. 107 (May) (2017) 109–119 https://doi.org/10.1016/j.cie.2017.03.012.
[23] Christian Kanzian, Franz Holzleitner, Karl Stampfer, Sarah Ashton, Regional energy wood logistics – optimizing local fuel supply, Silva Fenn. 43 (1) (2009), https://doi.org/10.14214/sf.464.
[24] Kalle Karttunen, Kari Väätäinen, Antti Asikainen, Tapio Ranta, The operational efficiency of waterway transport of forest chips on Finland's lake saimaa, Silva Fenn. 46 (3) (2012), https://doi.org/10.14214/sf.49.
[25] Krajnc, Wood Fuels Handbook, FAO, 2015, www.fao.org/3/a-i4441e.pdf.
[26] Hannes Lenz, Christine Idler, Eberhard Hartung, Ralf Pecenka, Open-air storage of fine and coarse wood chips of poplar from short rotation coppice in covered piles, Biomass Bioenergy 83 (December) (2015) 269–277 https://doi.org/10.1016/j.biombioe.2015.09.018.
[27] Lopez, Drying of Woof-Forest Chips Vol. XI (2012) Technical Backgrounds for

Advanced Techniques and Technologies in Biomass Production. biomassTradeCenter2 Project http://www.biomasstradecentre2.eu/wood-biomass-production/technical-backgrounds/.

[28] Juha Nurmi, The storage of logging residue for fuel, Biomass Bioenergy 17 (1) (1999) 41–47 https://doi.org/10.1016/S0961-9534(99)00023-9.

[29] M. Pettersson, T. Nordfjell, Fuel quality changes during seasonal storage of compacted logging residues and young trees, Biomass Bioenergy 31 (11–12) (2007) 782–792 https://doi.org/10.1016/j.biombioe.2007.01.009.

[30] Nazanin Shabani, Shaghaygh Akhtari, Taraneh Sowlati, Value chain optimization of forest biomass for bioenergy production: a review, Renew. Sustain. Energy Rev. 23 (July) (2013) 299–311 https://doi.org/10.1016/j.rser.2013.03.005.

[31] Nazanin Shabani, Taraneh Sowlati, A hybrid multi-stage stochastic programming-robust optimization model for maximizing the supply chain of a forest-based biomass power plant considering uncertainties, J. Clean. Prod. 112 (Part 4) (2016) 3285–3293 (January), https://doi.org/10.1016/j.jclepro.2015.09.034.

[32] W. Simpson, J. Tschernitz, Importance of Thichness Variation in Kiln Drying Red Oak Lumber, Forest Products Laboratory, U.S. Forest Service, 1979.

APPENDIX B

Integrating inbound and outbound transportation planning

Integrated planning of inbound and outbound logistics with a Rich Vehicle Routing Problem with Backhauls

Alexandra Marques* · **Ricardo Soares *†** · **Maria João Santos *†** · **Pedro Amorim*†**

Published in *Omega*, 2019. DOI: 10.1016/j.omega.2019.102172

*INESC TEC - Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência, Porto, Portugal
†Faculdade de Engenharia, Universidade do Porto, Porto, Portugal

Omega 92 (2020) 102172

Contents lists available at ScienceDirect

Omega

journal homepage: www.elsevier.com/locate/omega

Integrated planning of inbound and outbound logistics with a Rich Vehicle Routing Problem with backhauls☆

Alexandra Marques[a,*], Ricardo Soares[b], Maria João Santos[b], Pedro Amorim[b]

[a] INESC TEC - INESC Technology and Science, R. Dr. Roberto Frias, Porto 4200-465, Portugal
[b] INESC TEC, FEUP - Faculty of Engineering, University of Porto, R. Dr. Roberto Frias, Porto 4200-465, Portugal

ARTICLE INFO

Article history:
Received 9 January 2019
Accepted 16 December 2019
Available online 16 December 2019

Keywords:
Logistics planning
Vehicle routing with backhauls
Rich vehicle routing
Forest industry

ABSTRACT

This paper addresses the integration of the planning decisions concerning inbound logistics in an industrial setting (from the suppliers to the mill) and outbound logistics (from the mill to customers). The goal is to find the minimum cost routing plan, which includes the cost-effective outbound and inbound daily routes (OIRs), consisting of a sequence of deliveries of customer orders, pickup of a full truck-load at a supplier, and its delivery to the mill. This study distinguishes between three planning strategies: opportunistic backhauling planning (OBP), integrated inbound and outbound planning (IIOP) and decoupled planning (DIOP), the latter being the commonly used, particularly in the case of the wood-based panel industry under study. From the point of view of process integration, OBP can be considered as an intermediate stage from DIOP to IIOP. The problem is modelled as a Vehicle Routing Problem with Backhauls, enriched with case-specific rules for visiting the backhaul, split deliveries to customers and the use of a heterogeneous fleet. A new fix-and-optimise matheuristic is proposed for this problem, seeking to obtain good quality solutions within a reasonable computational time. The results from its application to the wood-based panel industry in Portugal show that IIOP can help to reduce total costs in about 2.7%, when compared with DIOP, due to better use of the delivery truck and a reduction of the number of dedicated inbound routes. Regarding OBP, fostering the use of OIRs does not necessarily lead to better routing plans than DIOP, as it depends upon a favourable geographical configuration of the set of customers to be visited in a day, specifically, the relative distance between a linehaul that can be visited last in a route, a neighboring backhaul, and a mill. The paper further provides valuable managerial insights on how the routing plan is impacted by the values of business-related model parameters which are set by the planner with some degree of uncertainty. Results suggest that increasing the maximum length of the route will likely have the largest impact in reducing transportation costs. Moreover, increasing the value of a reward paid for visiting a backhaul can foster the percentage of OIR in the optimal routing plan.

1. Introduction

The optimisation of the logistics processes has a whopping effect on improving the cost-efficiency of supply chains. Specifically, in forest-based supply chains, the inbound logistics bringing the wood from the forest to the mill can represent up to 30% of the total costs [6], while the outbound logistics bringing the wood-based products from the mill to the consumers can be equally high.

Despite recent studies showing that integrated planning of supply chain operations can lead to better results than decoupled planning [e.g., 2], inbound and outbound logistics planning are still dealt separately in most forest industries, as well as in other sectors. The complexity of the logistics operations, specificities of the transportation fleet and customer service levels are frequent justifications for this fact. In the wood-panel based industry, outbound logistics planning establishes the minimum-cost daily routes, henceforth called outbound routes (ORs), for delivering the ordered amounts of finished products to customers. This process accrues from the mill's production plan and impacts on the customer order lead time. Inbound logistics establishes the inbound routes (IRs), usually of a dedicated log-truck, consisting of a sequence of full truck-load trips between a wood sourcing location and the mill. The process is affected by wood procurement planning, ultimately impacting on the upstream forest harvest scheduling decisions. Similar transportation planning settings appear in the retail industry. Namely, in cases in which the retailer has the

☆ This manuscript was processed by Associate Editor Furini.

* Corresponding author.
E-mail addresses: alexandra.s.marques@inesctec.pt (A. Marques), ricardo.s@fe.up.pt (R. Soares), maria.j.santos@inesctec.pt (M.J. Santos), pamorim@fe.up.pt (P. Amorim).

https://doi.org/10.1016/j.omega.2019.102172
0305-0483/

option to pick-up products at suppliers besides just simply distributing to stores [47].

This paper studies the integration of inbound and outbound logistics in the context of the wood-based panel industry. The case study is driven from a real-life industrial application that operates on a multi-mill setting. The production strategy of the wood-based panels at each mill is Make-to-Order. The finished products are shipped to the customers in the day after its production. The stock of raw materials should be at least one week to overcome fluctuations in wood supply. The outbound logistics are planned locally, in the transportation department of each mill, while the inbound logistics are planned centrally, considering the bulk demand for all the mills. The goal here is to find daily minimum-cost outbound and inbound routes (OIRs) where the vehicle departing from each mill firstly performs a sequence of deliveries of the amounts ordered by the customers, and secondly, whenever is cost-effective, picks up a full truck-load of raw materials at a nearby supplier, and delivers it at the closest company's mill. OIRs allow better use of the delivery truck, when compared with ORs and further avoid dedicated IRs. This is possible because the driver can easily adapt the same truck that transported the wood boards with reinforcements in its structure so it can transport a full truck-load of wood chips. For wood-based supply chains, it is common that the inbound transport is carried in full truck-loads [e.g., [10,15,23]].

In this paper, the problem of finding OIRs is modelled as a Vehicle Routing Problem with Backhauls (VRPB). The VRPB is a variant of the well-known Vehicle Routing Problem (VRP) where the route visits several customers, in some performing deliveries (referred as linehauls) and in others pickups (the backhauls), but all deliveries must be made before any pickups [18]. In this study, we use the VRPB as a mean to tackle Integrated Vehicle Routing Problems, as outlined by Bektaş et al. [7], since the routing decisions related with the process of inbound logistics and those of the outbound logistics are dealt jointly. Moreover, there are essential business-related rules arising from our application to the wood-based panel industry that determine route feasibility, which are not yet fully covered in the VRPB literature and justify the formulation of a new variant of a rich VRPB, in line with the taxonomy proposed by Lahyani et al. [26].

The first set of business-related rules addressed in this study relate to the conditions determining the visit to a backhaul: (i) the backhaul can only be visited after all deliveries are performed, here called a precedence constraint, because the reinforcement of the truck for transporting the wood chips can only occur after the last delivery of the wood-based panels; (ii) there is at most one backhaul visited per route because the amount picked up is always a full truck-load since there are no wood availability constraints at suppliers; (iii) if there is a pickup at a backhaul it is mandatory that the same route includes its delivery at a mill. This is another type of precedence constraint ensuring that a mill is visited after a backhaul. However, operational practice indicates that the unloading mill may or may not be the mill of origin, because the company owns several mills geographically dispersed, and the truck can end the route in any of these mills, as long as the compatibility requirements between the types of raw materials available at the backhaul and accepted at the mill are accounted for; (iv) a backhaul may or may not be visited, which is known in the literature as selective backhauling; (v) routes without a backhaul are also feasible, and in this case, the route ends in the last linehaul visited, similarly to what occurs in an Open VRP (see Fig. 1). There are other studies on VRPB that work with precedence constraints and selectiveness. However, the possibility to optimise the decisions about visiting or not a backhaul and further choosing the delivering mill in order to minimise total logistics costs are new and important features of the problem under study. Another important case-specific rule determines that each customer may be

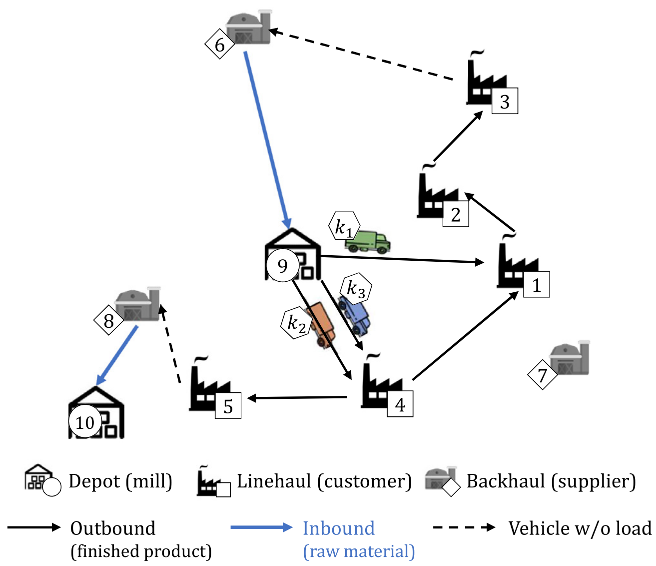


Fig. 1. Problem representation.

visited more than once by different vehicles, known in the VRP literature as *split deliveries*. The bundle of panels to be delivered at the linehaul customer is of variable size and weight. Therefore, several smaller bundles can be transported by the same truck, but larger bundles may need multiple trucks serving the same customer. Lastly, the available fleet is composed of trucks which are *heterogeneous* in terms of the transportation capacity. The transport is entirely outsourced to third-party carriers and paid based on a fixed daily use cost and a variable cost depending on the travelling distances of the 'for-hire' vehicles. We further emphasize that these business rules are also applicable in other industries besides the wood-panel one, such as in grocery retail.

The complexity of this real-world problem motivates a study about the main strengths and shortcomings of different inbound and outbound planning strategies, with greater or fewer degrees of integration. Furthermore, given the considerable size that these problems can achieve, it becomes relevant to envisage a scalable solution method, able to cope with the operational reality.

This research builds on a literature review on VRPB and other rich VRP variants with similarities to our problem. The first contribution of this paper is to develop a mathematical formulation to address a rich VRP that is primarily used to solve different planning strategies for obtaining OIRs. We apply it to a case study in the wood-based panel industry in Portugal and draw conclusions by comparing the routing plans obtained with those alternative planning strategies. Another contribution is to provide valuable managerial insights for planners about the impact of business-related model parameters over the optimal routing plan. Another contribution is to adapt the fix-and-optimise matheuristic presented by Sahling et al. [40] for obtaining good quality solutions for larger instances of this problem within a reasonable computational time.

The remainder of this paper is organised as follows. Section 2 provides a critical review of the literature regarding integrated transportation planning with a particular connection to the VRPB. This review covers extensions of VRPBs and solution methods developed to solve both artificial and real instances, and allows us to place our work in context. Section 3 presents the mathematical formulation of the three logistics planning strategies investigated in this work, namely the opportunistic backhauling, the integrated and the decoupled inbound-outbound transportation planning. Section 4 describes the solution approach developed, which is based on a fix-and-optimise algorithm. Section 5 presents the computational experiments performed with close-to-reality instances from a wood-based industry in Portugal. The routing plans obtained for the three planning strategies are compared, and relevant managerial insights are envisaged. The main conclusions are presented in Section 6.

2. Critical review of the state of the art

In the literature on logistics and transportation, the term integrated planning is broadly used to refer to situations where the routing decisions are tackled jointly with other decisions [43]. In some situations, the integration is between transportation decisions of different planning levels, for example, strategic decisions concerning the design of the transportation network and the tactical decisions related with the routes and assignment of the transport vehicles [e.g., 8]. In other situations, the integration is between the routing decisions and the decisions concerning other processes of the supply chain. The special issue by Bektaş et al. [7] on the integrated VRP shows examples of cases where vehicle routing is interlinked with decisions related to loading, production (or inventory), location, and speed optimisation. As an example, production-routing problems integrate production, products delivery (i.e., outbound logistics), and usually also inventory decisions [e.g., 1]. There are several examples in the forest literature where wood transportation to the mill (i.e., inbound logistics) and the upstream process of forest harvesting are planned jointly [e.g., 30].

As indicated by Speranza [43], a common feature of the studies on integrated transportation planning is that dealing with those decisions separately or hierarchically by solving the problems independently, leads to a sub-optimal solution for the integrated problem. In fact, integrated planning potentiates global efficiency gains, usually translated into cost savings. As an example, Archetti and Speranza [5] present significant savings of around 9.5% in terms of total cost and 9.0% in terms of the number of vehicles used when using a heuristic solution for an inventory-routing problem, in comparison with the solution obtained by sequentially and optimally solving the inventory management and the routing problems.

The main particularity of our study, not yet fully covered in the literature, is that the integration is between two processes of the supply chain - inbound and outbound logistics - wherein both processes the relevant decisions are related with the optimal vehicle routes. In fact, in our problem, it is the same vehicle that may perform both processes. There are significant differences in respect to the modelling approach because, in the other cases of integrated VRPs, such as production-routing, there are at least two types of decision variables, one for each process, and the correspondent linking constraints. While in ours, there are only the decision variables related to routing. The linkage between the two processes accrues from the way the routes are built.

The problem class that mostly resembles our problem is the VRPB, firstly introduced by Deif and Bodin [14]. Since then, there are several VRPB variants being studied in the framework of practical applications, as shown in the recent review of Koç and Laporte [24]. In general terms, the VRPB consists in finding the minimum cost routes, which start and end at the depot and visit a set of customers partitioned into linehauls (customers who require deliveries), and backhauls (customers who require pickups), all must be visited contiguously [e.g., 45].

The VRPB is not usually considered as an example of integrated vehicle routing planning. In fact, many of the industrial applications of the VRPB focus on the outbound logistics process, for example, in retail companies [e.g., 17,18]. In these cases, the route prioritises first all the products deliveries, and only afterwards the pickups, in order to attain a high vehicle utilisation. The customers are all of the same type (e.g., stores), but with different requirements (i.e., pickup or delivery) and the picked up material can be of a different type that cannot be mixed with the delivered products, such as empty boxes, damaged products or post-consumption material in reverse logistics. In other applications, such as the distribution of equipment to rentals [e.g., 16], or package delivery over a distribution network [e.g., 49], the inbound and outbound material is the same, and it is all planned together as a unique logistic distribution process.

Contrarily, we argue that our case study can be considered integrated transportation planning because the inbound and outbound logistics are two separate processes that nowadays are planned independently, involving different types of customers - i.e., suppliers of raw materials vs. consumers of finished products - sharing in common the depot/mill. Yano et al. [47] study a case resembling ours, in a retail chain with one centralized distribution centre, 40 stores and nearby vendors, where the route includes the delivery of goods to stores and the pickup of goods in nearby vendors. Planning includes dedicated routes for the vendors whenever it is not cost-efficient to include them in the delivery routes. The results of this work allowed savings in the order of a half-million dollars. With a similar strategy, Paraphantakul et al. [36] report a case-study in a cement industry, where cement customers are linehaul customers, and lignite mines are backhaul customers. The problem was solved using an ant colony optimisation method, and the company was able to save about 12% in the average tour duration.

The literature review on VRPB reveals examples of mathematical models, exact and heuristic methods for solving distinct problem variants. A general integer linear programming formulation and set partitioning formulation for the VRPB are presented in [24]. Among the most common extensions of VRPB found in the literature are the incorporation of time windows [20,25,33,39], multi-periods [13,33], multi-depots [11], heterogeneous fleet [28,41] and split deliveries [20,27,33,46]. There are also variants on the nature of the backhauling, such as the mixed VRPB that also allows deliveries to linehauls after pickups in backhauls [e.g., 48].

As the research on transportation planning advances more and more towards its practical application, several extensions of VRPs that consider real-life aspects of the logistics problems have emerged in the literature. The VRPs that cover such aspects, namely the integration of different logistics operations (e.g., inbound and outbound transport), the consideration of uncertainty or dynamism, or the inclusion of real constraints (e.g., time windows and multi-periodicity), fall into the vast class of Rich VRPs [9,26]. As our problem concerns a VRP with selective backhauls, heterogeneous fleet, and split deliveries, we can classify it as a rich VRPB. Table 1 presents a description of other VRPBs found in the literature that relate to our work, including the real-life aspects addressed in the problem and the respective types of solution methods used to solve the VRPB.

From Table 1, it is possible to observe that metaheuristics are the most popular methods used to solve VRPBs. This results from the fact that the VRPB is an NP-hard problem and, as such, very few exact methods are known to be efficient for large scale problems. Yano et al. [47] describe the problem using a set-covering formulation and then solve it using a procedure based on a Branch-and-Bound that starts from an initial solution obtained with simple heuristics. Gutiérrez-Jarpa et al. [21] introduce a Branch-and-Cut algorithm to solve a VRPB with split deliveries and test it in new problem instances adapted from the VRP instances with up to 100 customers, but only those instances with 50 customers or less can be solved to optimality. Davis et al. [13] use a commercial solver to find optimal transportation schedules that allow food banks to collect food donations from local sources and to deliver food to chari-

Table 1
Characteristics of the Rich VRPB under study and related works in the literature.

Reference	VRPB features							Solution method		
	TW	HF	SD	MD	MP	SB	MB	Exact	Metaheuristic	Matheuristic
[47]						•		•		
[39]	•								•	
[19]						•			•	
[21]	•		•			•		•		
[36]	•		•						•	
[25]	•	•							•	
[41]		•							•	
[28]		•	•						•	
[13]					•			•		
[11]				•					•	
[33]	•		•		•				•	
[35]	•	•					•	•		
[46]			•						•	
Our problem		•	•			•				•

Legend: TW (time-windows), HF (heterogeneous fleet), SD (split deliveries), MD (multi-depot), MP (multi-periodic), SB (selective backhauls), MB (mixed backhauls)

table agencies, through food delivery points. The problem is solved in two phases: first, the problem is formulated as a set-covering model to assign charitable agencies to food delivery points, and then, the problem is formulated as a VRPB enriched with constraints related to food safety, operator workday and collection frequency, also using the optimal solution of the first phase as an input. Oesterle and Bauernhansl [35] also study a logistic problem of a food company but considering a mixed VRPB with time windows, heterogeneous fleet, manufacturing capacity and driving time limits. The problem is formulated as a mixed integer programming model and also solved with a commercial solver in two phases. The first phase creates clusters of customers to visit, and at the second phase, the routes in each cluster are optimised.

With respect to metaheuristics, both local search and population-based methods have proved to be very efficient to deal with VRPB and its extensions. Examples of local search metaheuristics include tabu search [19,33], adaptive large neighborhood search [39], and variable neighborhood search [46]. Examples of population-based metaheuristics developed for the VRPB include ant colony optimisation [11,36] and evolutionary algorithms [25]. Moreover, two-phase heuristics are also investigated in the works of Salhi et al. [41] and Lai et al. [28].

Regarding matheuristic approaches, no references related to its adaptation to the VRPB were found. However, the literature accounts for several matheuristic approaches for various solving VRP variants. For example, the fix-and-optimise approach was initially proposed by Sahling et al. [40] for a lot-sizing problem, but it has been gaining recent interest in the literature for solving several rich routing problems with real-life aspects [e.g., 31]. This matheuristic consists in iteratively fixing different sets of binary variables from a mathematical model, thus allowing a commercial solver to only solve smaller parts of the global problem. Depending on the problem, the selection of the variables to be fixed or released needs to be carefully designed. Most references frame this approach in a variable neighbourhood decomposition search [22], where the number of variables to be released is progressively increased as a way to increase the neighbourhood sizes being explored [e.g., 12,42]. Other research works use distinct heuristic concepts, such as tabu search [e.g., 38] by using a tabu list for the variables being fixed.

Our work is distinct from the ones revisited in this section. It contributes to the literature because it not only describes a new formulation for a rich VRPB that can be used to address different transportation planning strategies but also investigates a fix-and-optimise method to solve the problem, which was not yet addressed in VRPB literature.

3. Problem formulation

This section outlines the main planning strategies for the integration of inbound and outbound logistics processes, which will be addressed in this paper. For each one of these planning strategies, mathematical formulations will be provided, which will be the basis for the sections that follow.

3.1. Logistics planning strategies

The integration of inbound and outbound logistics by finding the optimal OIRs can be staged in two distinct planning strategies, in opposition to a simpler strategy of decoupled planning, similar to what is used today by the company:

- Opportunistic backhauling planning (OBP): In this strategy, the primary process to be considered is the outbound logistics. The outbound transportation plan encompasses ORs and cost-effective OIRs, but another plan exists for IRs. There is an underlying idea that OIRs can provide only a residual amount of the raw materials demanded and IRs assure the vast majority of the demand.
- Integrated Inbound and Outbound Planning (IIOP): In this strategy, both processes of inbound and outbound logistics are planned jointly. The transportation plan encompasses all types of routes – ORs, OIRs and IRs.
- Decoupled Inbound and Outbound Planning (DIOP): This strategy implies that both processes of inbound and outbound logistics are planned independently. The outbound transportation plan (or delivery plan) encompasses the ORs, while the inbound plan (or supply plan) encompasses IRs, there are no OIRs. In the current situation of the case study, logistics planning occurs in separate company departments. IRs are planned centrally and ORs are planned in a department at each mill.

From the point of view of process integration, OBP can be considered an "intermediate" stage, from DIOP towards IIOP, as well as from the point of view of the level of organisational changes needed for its adoption. In fact, OBP impacts mostly on the planners of the outbound logistics in each mill and on the truck drivers while IIOP implies a major restructuring from merging (and possibly centralising) the inbound and outbound logistics planning departments. From a modelling point of view, the mathematical formulation for OBP and IIOP are similar. For the purpose of simplification, this section focuses on OBP, making the necessary adjustments to IIOP afterwards. The section ends with the description of DIOP.

3.2. Opportunistic backhauling planning (OBP)

OBP can be modelled as a rich, capacitated Vehicle Routing Problem with selective backhauls and split deliveries. Considering a set of mills M, a set of linehaul customers L whose demand needs to be fulfilled, and a set of suppliers backhauls B with raw materials available for the mills that may or may not be visited. The problem consists in finding the optimal daily minimum-cost routes for a set of trucks K, starting at the mill, encompassing one or many deliveries to linehauls, and including at the most one pickup of a full truck-load of a given type of raw materials at a backhaul, which is selected based on the best fit with one of the possible destination mills. The set of types of raw materials to be collected at a backhaul is represented by set P. Hence, the problem components include:

- the fleet of $|K|$ trucks, where each truck $k \in K$ has a given capacity (Q_k) and can perform both deliveries and pickups. There is a fixed cost for the daily usage of a vehicle (f^k) and a variable cost (c_{ij}^k) proportional to the travelled distances;
- the $|M|$ mills owned by the company that are geographically dispersed. Each mill $m \in M$ receives wood chips and produces wood-based panels on a make-to-order basis. The fleet is assigned to a specific mill or origin (or depot), from where the routes start. According to operational practice, in case of a route with a backhaul, the truck can unload the raw materials in any of the company's mills, which may or may not be the mill of origin. There is a minimum amount of raw materials to be backhauled to all mills (β);
- the $|L|$ linehaul customers that are characterized by a given demand of a finished product, which must be fulfilled (q_l) at each linehaul $l \in L$. Split deliveries can occur, meaning that each customer may be visited more than once (each visit consisting in at least a ψ amount), but each truck may visit a customer at most once;
- the $|B|$ backhaul suppliers that are also geographically dispersed. Also, according to the operational practice, it is assumed that all have unlimited availability, hence pickups correspond to full-truck loads. The type of raw materials that are available may also vary amongst them;
- the $|P|$ types of raw materials consisting of wood chips of variable size and moisture content, sawdust and recycled wood. Some types of raw materials are more desirable to the mills than others. There are also compatibility issues with respect to the types of raw materials available and demanded at the different locations.

Contrarily to other VRPs found in the literature, the time window constraints related to the earliest or latest time to arrive at each location are not of importance. However, the maximum distance travelled in a route is limited by a parameter α. It is noteworthy that the route length can be constrained in terms of travelling time, to account for driving time regulations stating maximum driving or working times. However, in this case, the value of the maximum distance travelled was set with the planner as an average of the actual routes length, already implicitly considering all the necessary stops, hence simplifying problem modelling. In summary, the characteristics of the feasible routes are: (i) start at a home depot with the truck loaded up to its maximum capacity, with the products ordered by the linehaul customers; (ii) perform a sequence of deliveries to the linehauls; (iii) if it is cost-effective and doable during the maximum route length, the vehicle travels empty to a nearby backhaul supplier to pick up a full truck-load of raw materials to be delivered at any of the company's mill, where the route ends (specific to OIRs); and (iv) if a backhaul is not visited, the route is ended when the truck is empty after visiting the last linehaul of the route (specific to ORs), as the company does not pay for trips where the truck does not transport merchandise.

3.2.1. Modelling approach

The rVRPB under study is modelled as a graph $G = (V, A)$ where V is the set of all vertices, $V = \{0\} \cup L \cup B \cup M$ and A is the set of all possible arcs. We adopt a standard flow VRP formulation with 3-index decision variables x_{ij}^k equal to 1 if vehicle $k \in K$ travels from customer $i \in V$ to $j \in V$ and zero otherwise. Like in the standard VRPB formulation proposed by Parragh et al. [37], we distinguish the vertices in linehauls and backhauls, in order to model the precedence constraints.

However, the typical VRPB constraints assuring that each vertex is visited exactly once do not apply, due to the possibility of selective backhauls (i.e., backhauls may or may not be visited) and the split deliveries at the linehauls (i.e., linehauls are visited more than once).

To avoid the complexity of a multi-depot and open VRP, we propose a *2-echelon backhauls network*, starting and ending at the same fictitious depot 0. In fact, when the route starts, the fictitious depot corresponds to the mill of origin from where the customers' orders will be delivered. Since there is a fleet dedicated to each mill when the route starts, routing planning for each mill can be done separately as a single depot. When the route ends, the fictitious depot corresponds to a fictitious location whose distance from the last vertex visited in the route is equal to zero. Hence, the 2-echelon backhauls network is composed by the first echelon of backhauls corresponding to the suppliers and the second echelon of backhauls corresponding to the mills to be supplied by the backhauled amounts. Additional constraints are needed to assure that a mill can only be visited after a backhaul (see Fig. 2).

The decisions whether a backhaul is visited in a route or not, and if so, to which mill to go next, are based on a new parameter related with the *reward paid for visiting that backhaul and a mill next* (δ_{bm}). Like in previous studies of VRP with selective pickups [e.g., 19] and other formulations of VRP with profits [e.g., 4], the reward is used to make an arc linehaul to backhaul more or less attractive. The reward corresponds to a payment per each ton of raw materials picked-up in a backhaul and delivered in a neighbouring mill. If the route ends after visiting the last linehaul, then there is no positive reward associated with that route. Hence, the reward parameter is used in the objective function, which trades-off between the sum of the travelling costs for visiting the backhaul after the last linehaul and moving from there to a mill, and the reward gained for visiting that backhaul. The reward parameter is also used to address compatibility issues related to the type of raw material p to be transported from a given backhaul b to a given mill m. In fact, if p is not available in b or not accepted in m then $\delta_{bm} = 0$. On the contrary, if there are several types of raw materials that can be transported from b to m, the value of δ_{bm} corresponds to the value of the most profitable material because there are no other aspects determining the choice between them. Consequently, the set P does not need to be considered in this model. However, in other real-life applications where the availability at the backhauls and or demand at the mills is limited and varies per type of product, the set P should be properly incorporated in the model, leading to a four-index decision variable x.

A new decision variable is needed to assure that, despite the possibility of *splitting the deliveries* to a linehaul, each delivery cannot exceed the truck capacity and that the total amount delivered in the several routes that visit it meets the expected demand. Previous studies used continuous variables w_i^k representing the quantity transported by vehicle $k \in K$ to/from customer $i \in V$ for a similar purpose [e.g., 34]. However, in the rVRPB under study, without time windows, these variables are insufficient for sub-tour elimination. In this context, a new set of continuous variables u_{ij}^k rep-

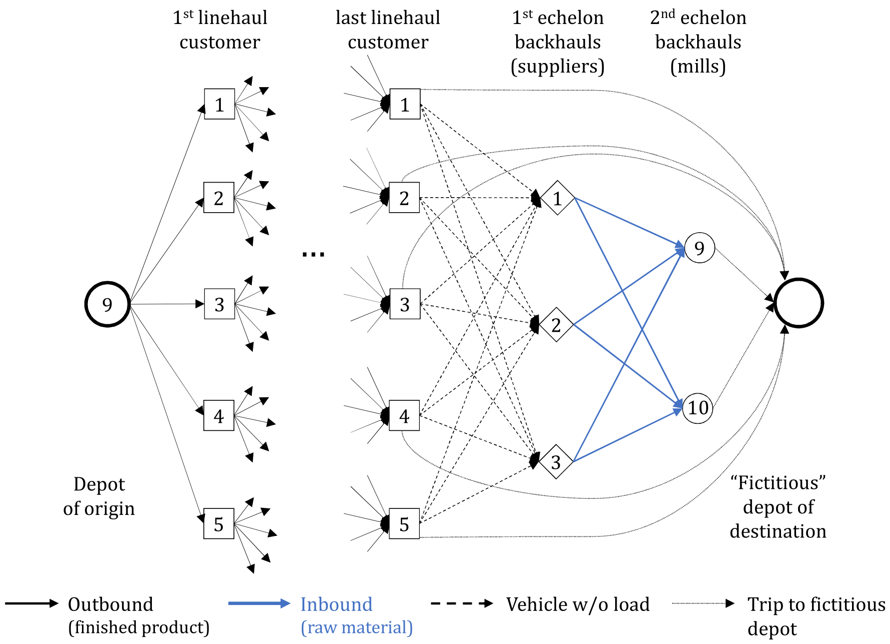
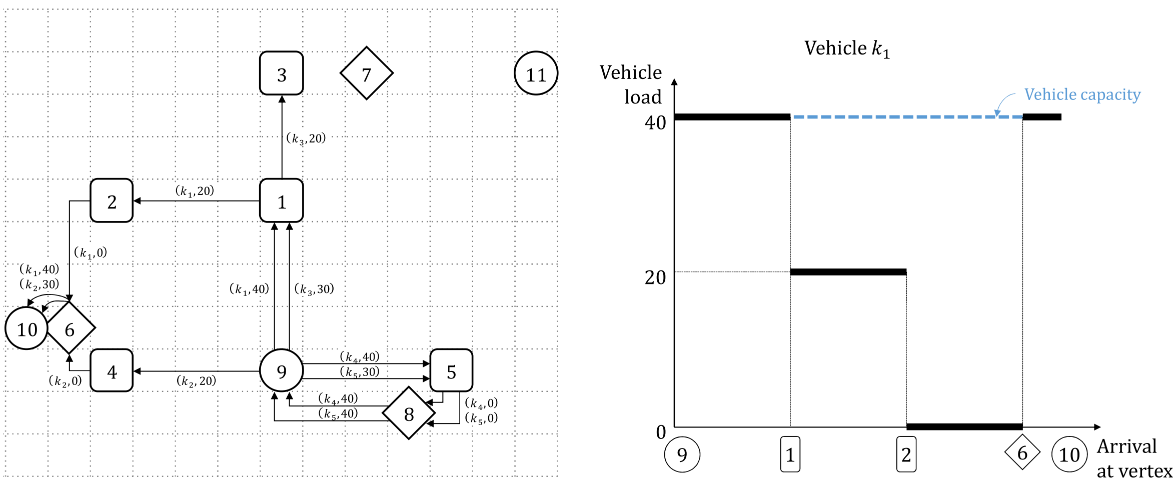


Fig. 2. Network representation of the problem.



(a) feasible solution for vehicles k_1, k_2, k_3, k_4, k_5 (b) variation of the load of vehicle k_1 along the route

Fig. 3. Example of a feasible solution for a rVRPB.

resent the load of vehicle $k \in K$ when traversing arc $(i, j) \in A$. Variables u_{ij}^k are a natural adaptation of variables u_i [7,44] to a multi-route and split delivery situation. Additional constraints are needed to account for the routes with backhauls. In this case, the truck-load is higher before visiting the first linehaul, then progressively decreases until reaching zero after visiting the last linehaul. If a backhaul is visited, the pickup corresponds to a full truck-load. As an example, for a given route k, encompassing {0, i, i', i'', j, 0}, where i, i', $i'' \in L$ and $j \in B$, then the following rules apply: $u_{0i}^k \leq u_{ii'}^k \leq u_{i'i''}^k, u_{i''j}^k = 0, u_{j'0}^k = Q_k$.

Fig. 3 exemplifies a feasible solution for the OBP starting in the node 9, in a network composed by 5 linehauls (numbered 1 to 5), 3 backhauls (numbered 6 to 8) and 3 mills (numbered 9 to 11). For simplification purposes, only the arcs used in the solution are represented in Fig. 3a. The demand (in ton) at the linehauls is $q_1 = 30, q_2 = 20, q_3 = 20, q_4 = 20, q_5 = 70$. The reward for visiting a backhaul is 0.1€ /ton in all cases. The available fleet is composed of 5 trucks, with capacity (in ton) $Q_1 = 40, Q_2 = 30, Q_3 = 30, Q_4 = 40, Q_5 = 40$. The linear distances between vertices (d_{ij}) are computed in reference to the background grid with 1km by 1 km, for example, $d_{13} = 2$ km. The fixed cost for using a vehicle is zero, and the variable cost is 1 € /km.

The routing plan foresees the use of all five vehicles: k_1, k_2, k_4 and k_5 are OIRs while k_3 is an OR ending after visiting linehaul 3. There are split deliveries in linehauls 1 and 5. Total costs are 29€ and total revenues are 15€. The values of u_{ij}^k for truck 1 are shown in Fig. 3b.

This example is instrumental in showing the impact of the reward value over the final routing solution. In fact, the route visiting linehaul 4 will always visit backhaul 6, and then mill 10, because the extra cost for visiting this pair backhaul-mill is 1€ ($d_{4,6} = 1e, d_{6,10} = 0 \Rightarrow c_{4,10}^k = 1$) and the minimum revenue is 3€ ($\delta_{bm} = 0.1$€ /ton, $\min\{Q_k\} = 30$ ton $\Rightarrow u_{6,10}^k \geq 30, \forall k \in K, \delta_{6,10} = 0.1$. Applying a similar logic, it is expected that the route visiting linehaul

3 will visit backhaul 7 if $\delta_{7,11} \geq 0.4$, since the extra cost for visiting the backhaul and mill is 4€ and $u^k_{7,11} \geq 30, \forall k \in K$.

3.2.2. Mathematical formulation

For the sake of convenience, before presenting the mathematical formulation, we resume the necessary decision variables, sets and parameters.

Decision variables:

x^k_{ij} $\begin{cases} 1 & \text{if vehicle } k \text{ travels from location} i \text{ to } j; \\ 0 & \text{otherwise.} \end{cases}$

u^k_{ij} load of vehicle $k \in K$when traversing arc $(i, j) \in A$

Sets:

L set of linehauls (customers where finished products are delivered)

B set of backhauls (suppliers where raw materials can be picked up)

M set of mills (where raw materials are delivered if a backhaul is visited)

V set of vertices; $V = \{0\} \cup L \cup B \cup M$

K set of vehicles

Parameters:

q_i quantity to be delivered to customer $i \in L$(ton)

c^k_{ij} cost of transportation with vehicle $k \in K$ from$i \in V$to $j \in V(e)$

f^k fixed cost of using vehicle$k \in K$ in a daily route (e)

Q_k transportation capacity of vehicle $k \in K$ (ton)

d_{ij} travelling distance from $i \in V$ to $j \in V$(km)

α maximum distance travelled in a route (km)

β minimum amount of raw materials to be backhauled (ton)

δ_{bm} reward for picking up one unit of raw material at backhaul $b \in B$ and delivering it to mill $m \in M(e)$

ψ minimum amount of order delivered to a linehaul (ton)

Model [P0]

$$\min \sum_{k\in K} \sum_{j\in V\setminus\{0\}} f^k x^k_{0j} + \sum_{k\in K} \sum_{i\in V} \sum_{j\in V} c^k_{ij} x^k_{ij} - \sum_{k\in K} \sum_{i\in B} \sum_{j\in M} \delta_{ij} u^k_{ij} \tag{1}$$

subjected to:

$$\sum_{i\in V} x^k_{ij} \leq 1 \qquad \forall j \in L \cup B, \forall k \in K \tag{2}$$

$$\sum_{i\in B} \sum_{j\in B} \sum_{k\in K} x^k_{ij} = 0 \tag{3}$$

$$\sum_{i\in V\setminus B} \sum_{j\in M} \sum_{k\in K} x^k_{ij} = 0 \tag{4}$$

$$\sum_{i\in M} \sum_{j\in V\setminus\{0\}} \sum_{k\in K} x^k_{ij} = 0 \tag{5}$$

$$\sum_{i\in B} \sum_{j\in L\cup\{0\}} \sum_{k\in K} x^k_{ij} = 0 \tag{6}$$

$$\sum_{j\in B} \sum_{k\in K} x^k_{0j} = 0 \tag{7}$$

$$\sum_{j\in L} x^k_{0j} = \sum_{i\in L\cup M} x^k_{i0} \qquad \forall k \in K \tag{8}$$

$$\sum_{i\in V} x^k_{ij} = \sum_{i\in V} x^k_{ji} \qquad \forall j \in V, \forall k \in K \tag{9}$$

$$u^k_{ij} \leq Q_k x^k_{ij} \qquad \forall (i, j) \in A, \forall k \in K \tag{10}$$

$$\sum_{i\in V} u^k_{ij} - \sum_{i\in V} u^k_{ji} \geq \psi \sum_{i\in V} x^k_{ij} \qquad \forall j \in L, \forall k \in K \tag{11}$$

$$\sum_{i\in L} \sum_{j\in B\cup\{0\}} \sum_{k\in K} u^k_{ij} = 0 \tag{12}$$

$$\sum_{i\in V} \sum_{k\in K} \left(u^k_{ij} - u^k_{ji}\right) = q_j \qquad \forall j \in L \tag{13}$$

$$\sum_{i\in V} \sum_{j\in V} d_{ij} x^k_{ji} \leq \alpha \qquad \forall k \in K \tag{14}$$

$$\sum_{i\in B} \sum_{j\in M} \sum_{k\in K} u^k_{ij} \geq \beta \tag{15}$$

$$x^k_{ij} \in \{0, 1\}, u^k_{ij} \geq 0 \qquad \forall (i, j) \in A, \forall k \in K \tag{16}$$

The objective function (1) minimizes the total costs, decomposed into fixed costs (proportional to the number of vehicles used) and the variable costs (proportional to the total travelled distance), decreased by the revenue obtained for visiting backhauls and mills in the course of the OIR. Constraints (2) assure that any location can be visited at most once by each truck. Regardless, each linehaul and backhaul can be visited by several routes. Constraints (3)–(7) deal with route precedence rules, resulting from the specificities of this rVRPB for the wood-based panel industry. Specifically, constraints (3) state that the transport from a backhaul to another backhaul is not possible. Constraints (4) assure that the mill can only be visited after a backhaul. Constraints (5) assure that after visiting a mill, the only possibility is to go to the ending depot. Constraints (6) state that after visiting a backhaul, the next visit cannot be to a linehaul nor to the depot. Constraints (7) assure that the route cannot visit a backhaul after the depot. Constraints (8) and (9) are the typical VRP flow conservation constraints, at the depot and at each vertex, respectively. Constraints (10) are linking constraints, assuring that there is only a given amount transported to/from the customer if the customer is visited. Constraints (11) to (13) assure the elimination of sub-tours. Specifically, constraints (11) assure that the load of trucks progressively decreases as it visits the linehauls, and the amount delivered should be higher than a minimum amount. By considering the lower bound of the minimum amount, the model avoids undesirable solutions where $x^k_{ij} = 1$ and $u^k_{ji} - u^k_{ij} = 0$, which may occur for example if a linehaul (i') is visited in the course of a route from i to j, i.e., $x^k_{ii'} = x^k_{i'j} = 1$ (instead of $x^k_{ij} = 1$) but the amount delivered in i' is zero $\left(u^k_{i'i} - u^k_{ii'} = 0\right)$ due to the fact that the distance matrix does not obey to the triangular inequality (i.e., $\exists\, d_{ij} : d_{ij} > d_{ii'} + d_{i'j}$). Constraints (12) state that the truck leaves empty after visiting the last linehaul and constraints (13) assure that the demand at the linehauls is completely fulfilled. Constraints (13) together with constraints (2) account for the possibility of split deliveries at the linehauls. Constraints (14) assure that the maximum allowable distance of the daily route cannot be exceeded. It is noteworthy that if the maximum route length is constrained by the time travelled, then, this would require another type of auxiliary variables to count the route duration and consequent changes in these constraints, with similarities with other VRPs with time windows [e.g., 44]. Constraints (15) set a minimum amount of raw materials to be backhauled to mills. Finally, constraints (16) determine the domain of the decision variables.

3.2.3. Special situation in which the rVRPB is simplified to a rich capacitated VRP

A problem variant of the rVRPB consists in removingconstraints (14) and (15). In this situation, where there is no limitation to the route length and there is no minimum backhauling amount, a backhaul will be visited whenever it is cost-effective, according to the trade-off between the extra transportation cost (from travelling from the last linehaul, to that backhaul and to its closest mill) and the revenue (associated with delivering the load from the backhaul to the closest mill). From a modelling perspective, this means that, knowing which is the last visited linehaul in a route, it is possible to compute beforehand if and which backhaul and mill should be visited to minimize total costs. Consequently, the mathematical model can be simplified to a Rich Capacitated VRP (rCVRP) with split deliveries. This problem will only consist in sequencing the linehauls to be visited in each route, thus determining which linehaul will be last in each route.

This adaptation relies on a data pre-processing procedure (described in Algorithm 1) which consists in computing the minimum cost of having a given linehaul visited last in a vehicle route. If the cost of visiting a backhaul at the end of the route is lower than finishing the route at the depot (line 5), the cost associated with the arc heading to the depot is updated to the summed costs of pickup at the backhaul, delivering to the mill and returning to the depot, subtracted by the corresponding reward for performing the delivery to that mill (line 6). All combinations of vehicles, linehauls, backhauls, and mills are tested in this pre-processing stage, therefore ensuring that the vehicle arcs heading to the depot account for the minimum possible cost, which either corresponds to performing backhauling at the most advantageous locations or finishing its route after visiting the last linehaul. Finally, the sets of backhauls and mills are removed from the problem.

Algorithm 1: Data pre-processing for adapting the rVRPB to a rCVRP.

1 **foreach** *vehicle* k **in** K **do**
2   **foreach** *linehaul customer* j **in** L **do**
3   **foreach** *backhaul customer* i **in** B **do**
4   **foreach** *mill customer* m **in** M **do**
5   **if** $c^k_{ji} + c^k_{im} + c^k_{m0} - \delta_{im} \cdot Q_k < c^k_{j0}$ **then**
6           $c^k_{j0} := c^k_{ji} + c^k_{im} + c^k_{m0} - \delta_{im} \cdot Q_k$;
7 $V := V \setminus (B \cup M)$; $B := \varnothing$; $M := \varnothing$;

Afterwards, the new model for the rCVRP can be built upon [P0] by changing the objective function and removing constraints related with the sets of backhauls and mills, as shown in model [P1].

Model [P1]

$$\min \sum_{k\in K}\sum_{j\in V\setminus\{0\}} f^k x^k_{0j} + \sum_{k\in K}\sum_{i\in V}\sum_{j\in V} c^k_{ij}x^k_{ij} \tag{1b}$$

subjected to (2), (8)–(13) and (16) of model [P0]

3.3. Integrated inbound and outbound planning (IIOP)

As stated before, the IIOP strategy consists in jointly planning all types of routes, including OIRs, ORs only for delivery of finished products and IRs for pickup of raw materials. Model [P2] for IIOP can be built upon adaptations of [P0], that account for the IRs, as follows. Constraints (7) are removed to allow dedicated routes from the depot to a backhaul. A new parameter δ^D_{bm} represents the reward for picking up one unit of raw material at backhaul $b \in B$ and delivering it to mill $m \in M$ (€) in the course of the dedicated route. A new set of auxiliary continuous variables y^k_{ij} is needed to represent the amount picked up in $b \in B$ and delivered in mill $m \in M$ by vehicle $k \in K$ in a direct route. The objective function (1c) is adapted accordingly. A new set of constraints (17) defines variables y^k_{ij} and constraints (18) set its bounds. Considering an arc (i, j), $i \in B, j \in M$, with $x^k_{ij} = 1$, if $x^k_{0i} = 1, i \in B$, then k is in a dedicated route, and according to the conjugation of constraints (17) and (18), $y^k_{ij} = u^k_{ij}$. If $x^k_{0i} = 0, i \in B$, then k is in an OIR, and $y^k_{ij} = 0$.

Model [P2]

$$\min \sum_{k\in K}\sum_{j\in V\setminus\{0\}} f^k x^k_{0j} + \sum_{k\in K}\sum_{i\in V}\sum_{j\in V} c^k_{ij}x^k_{ij} - \sum_{k\in K}\sum_{i\in B}\sum_{j\in M} \delta^D_{ij} y^k_{ij} - \sum_{k\in K}\sum_{i\in B}\sum_{j\in M} \delta_{ij}(u^k_{ij} - y^k_{ij}) \tag{1c}$$

subjected to (2)–(6), (8)–(16) of model [P0] and

$$y^k_{ij} \le Q_k x^k_{ij} \qquad \forall i \in B, \forall j \in M, \forall k \in K \tag{17}$$

$$y^k_{ij} \le u^k_{ij} - (1 - x^k_{0i})Q_k \qquad \forall i \in B, \forall j \in M, \forall k \in K \tag{18}$$

$$y^k_{ij} \ge 0 \qquad \forall i \in B, \forall j \in M, \forall k \in K \tag{19}$$

3.4. Decoupled inbound and outbound planning (DIOP)

As stated before, DIOP corresponds to the planning strategy currently used, where the ORs and IRs are planned independently and there are no OIRs. For the outbound logistics planning, the optimal ORs can be obtained by solving model [P3] that is an adaptation of model [P0], considering the nonexistence of backhauls and mills. For the inbound planning, the optimal IRs can be obtained by solving a model [P4], also an adaptation of model [P0], acknowledging only the routes from the depot/mill of origin to the backhauls.

Model [P3]

$$\min \sum_{k\in K}\sum_{j\in V\setminus\{0\}} f^k x^k_{0j} + \sum_{k\in K}\sum_{i\in V}\sum_{j\in V} c^k_{ij}x^k_{ij} \tag{1d}$$

subjected to (2), (8)–(11), (13)–(14) and (16) of model [P0]

Model [P4]

$$\min \sum_{k\in K}\sum_{j\in V\setminus\{0\}} f^k x^k_{0j} + \sum_{k\in K}\sum_{i\in V}\sum_{j\in V} c^k_{ij}x^k_{ij} - \sum_{k\in K}\sum_{i\in B}\sum_{j\in M} \delta^D_{ij} u^k_{ij} \tag{1e}$$

subjected to (2)–(6), (8)–(10) and (14)–(16) of model [P0]

4. Solution approach

4.1. Fix-and-optimise approach

As stated in the literature review, the complexity of the VRP problems in real-life applications justifies the use of matheuristics. In this study, all the different models presented before are solved with a fix-and-optimise (F&O) approach in case of the large instances (i.e., more than 30 customers). This solution method was firstly presented by Sahling et al. [40] for lot-sizing problems, but has been successively used for solving complex routing problems with promising results [e.g., 29,32].

The F&O matheuristic approach consists in iteratively solving several smaller mixed integer programming (MIP) sub-problems of the original model. The design of each sub-problem is problem-dependent and the obtained results highly depend on its adequate design. In this approach, we define a sub-problem as a set of decision variables to be either released or fixed in the original MIP model. Fixing a variable consists in setting its lower and upper bounds to the current solution value, thus precluding it from being changed in a solver iteration. On the other hand, releasing a variable consists in restoring a fixed variable to its original lower and upper bound values. For the problem at hand, two distinct sub-problem types were conceived, named RouteRelease and LocationRelease.

The RouteRelease sub-problem releases all decision variables associated with a given set of routes in the incumbent solution, based on proximity criteria of these routes. Route proximity is defined by the centroids of each route, which are computed as the non-weighted averages of the location coordinates that are visited. The outline of the RouteRelease sub-problem construction procedure is illustrated in Algorithm 2. The procedure starts by computing the centroid of each route in the incumbent solution (lines 2–4). For unused vehicles, the route's centroid is given by the depot's coordinates. A pivot route is selected at random from the incumbent solution (line 5), after which all other routes are ordered by its centroid's distance to the pivot route (line 6). The n routes with the lowest distance to centroid of the pivot route are then released in the sub-problem (lines 7–12).

Algorithm 2: Route Release sub-problem construction.

input: vars (MIP model routing decision variables),
sol (incumbent solution),
n (number of routes to be released in the subproblem)

1 released_routes = ∅;centroid_list = ∅;
2 **foreach** route **in** sol **do**
3   **compute** *centroid* **of** route;
4   **append** *centroid* **of** route **to** centroid_list;
5 rt ← *random route*; cnt ← *centroid of* rt;
6 **order** centroid_list **by** *descending order* **of** *their distance to* cnt;
7 released_routes ← n *first routes* **in** centroid_list;
8 **foreach** var **in** vars **do**
9   **if** var *is associated with a vehicle* **in** released_routes **then**
10   **release** var;
11   **else**
12   **fix** var;

The LocationRelease sub-problem consists in releasing a given set of linehaul locations based on its geographical proximity. The procedure is described in Algorithm 3, and it starts by selecting a pivot linehaul (line 1), after which we retrieve all routes in the incumbent solution where the pivot linehaul is visited. Afterwards, we retrieve all the additional linehauls that are visited in these routes (lines 4–5). Finally, the n closest linehauls to the pivot linehaul that were previously selected are released (lines 8–14).

The overall structure of the matheuristic is shown in Algorithm 4.

The solution method requires an initial solution s_0 with objective function f_0, which is obtained through a greedy nearest neighbour heuristic (line 1): we select a random vehicle and construct its route by visiting the nearest unsatisfied linehaul until vehicle capacity is exhausted. The process is repeated until all linehaul demand is satisfied. No routes to backhauls are considered in the constructive phase.

After obtaining an initial solution, the matheuristic is then started. To that effect, sub-problem construction is initiated, whose size is controlled through the general principles of a Variable Neighbourhood Decomposition Search (VNDS), similar to what is presented in [22]. Sub-problems are constructed in line 7, after which the MIP model is fed the incumbent solution s_{cur} and the sub-problem is solved by a MIP solver (lines 8–9).

After each solver iteration, the obtained solution s_{solve} is evaluated against the incumbent solution (lines 10–15). If the obtained

Algorithm 3: Location Release sub problem construction.

input: vars (MIP model routing decision variables),
sol (incumbent solution),
n (number of locations to be released in the subproblem)

1 released_locations = ∅;candidates = ∅;loc ← *random linehaul*;
2 **foreach** route **in** sol **do**
3   **if** route **traverses** loc **then**
4   **foreach** linehaul **in** route **do**
5   **append** linehaul **to** candidates;
6 **order** candidates **by** *descending order* **of** *their distance to* loc;
7 released_locations ← n *first locations* **in** candidates;
8 **foreach** var **in** vars **do**
9   **if** var *is associated with a linehaul* **in** released_locations **then**
10   **release** var;
11   **else if** var *is associated with a mill* **then**
12   **release** var;
13   **else**
14   **fix** var;

Algorithm 4: Matheuristic outline.

input: MIPmodel (mixed integer programming model),
P (list of possible sub-problems to be used),
n_p (initial neighbourhood size for sub-problem p),
N_p (maximum neighbourhood size for sub-problem p),
$I_{p,n}$ (limit of consecutive non-improvement iterations for sub-problem p of size n),
$TL_{p,n}$ (time limit for solver iterations of sub-problem p of size n),
imp (minimum solution improvement to reset the no-improvement counter i)

1 s_0, f_0 = nearest_neighbour();
2 $s_{cur} = s_0$; $f_{cur} = f_0$; i = 0;
3 p = "RouteRelease";
4 **while** *termination criteria not met* **do**
5   n = n_p;
6   **while** n ≤ N_p **do**
7   **construct** sub-problem of type p **with** size n;
8   **feed** MIPmodel with initial solution s_{cur};
9   s_{solve}, f_{solve} = MIPsolve(MIPmodel , $TL_{p,n}$);
10   **if** $f_{solve} < f_{cur} -$ imp **then**
11             $s_{cur} = s_{solve}$; $f_{cur} = f_{solve}$; i = 0; p = "RouteRelease";
12   **break**
13   **else**
14   **if** $f_{solve} < f_{cur}$ **then**
15                 $s_{cur} = s_{solve}$; $f_{cur} = f_{solve}$;
16   i = i + 1;
17   **if** i > $I_{p,n}$ **then**
18   i = 0;
19   **if** n = N_p **then**
20   **if** p = "LocationRelease" **then**
21   p = "RouteRelease";
22   **else** p = "LocationRelease";
23   **break**
24   **else increase** n;
25 **output** s_{cur}, f_{cur}

solution did not yield an improvement of at least imp (line 10), we consider this a non-improvement iteration and increment the non-improvement counter. Nevertheless, we will accept the obtained solution even if it is not significantly better than the previous one (line 15). After a given number of consecutive non-improvements, the VNDS framework takes place either by increasing sub-problem size or switching the sub-problem type, if the current sub-problem size has been maxed out (lines 16–24). In the occurrence of a significant improvement of the problem's objective function, sub-problem type is re-set to `RouteRelease` and its initial size (line 11).

The matheuristic approach always initializes with the `RouteRelease` sub-problem type and the `LocationRelease` sub-problem is used after a significant number of non-improvements of the `RouteRelease` sub-problem. This algorithmic structure was conceived by bearing in mind that `RouteRelease` would be used as a more disruptive sub-problem, which would explore more disperse sections of the solution space, while the `LocationRelease` sub-problem focuses more on intensification.

4.2. Data pre-processing

Pre-processing the instance data related to the network generation is a common procedure in VRPs [e.g., 37,42] to simplify the mathematical formulation and achieve better performance in the optimisation solver. The pre-processing procedure used prior to solving the models is threefold. The sub-set of arcs to be considered is presented in Table 2.

First, we remove all the arcs that lead to an unfeasible route, i.e., arcs that violate the precedence constraints (3) to (7). Second, we eliminate all arcs from linehauls to backhauls where its visit is not economically worthwhile, according to the given reward for visiting a backhaul. These arcs are only generated if they respect the condition exhibited in line 5 of Algorithm 1.

Third, arcs from backhauls are only generated to its closest mill, as delivering merchandise to more distant mills will only induce an increase of the problem's objective function.

It should be noted that this data pre-processing procedure does not cut off optimal solutions only if we do not impose a minimum inbound quantity to be collected from backhauls via constraints (15). If this is not the case, this procedure may induce sub-optimality or even turn the model infeasible because there are no cost-effective backhauls to visit. Therefore, in these situations, a trade-off between optimality and simplicity must be taken into account.

5. Computational experiments

The proposed approach was applied in a case study in a wood-based panel company in Portugal. The mathematical model was implemented in Gurobi 7.5 commercial solver. The solution method was developed in Python 3.6. The mathematical models were subject to the data pre-processing procedure described earlier and used to compare the gains of the IIOP strategy with the DIOP one, which is currently done by the company. A set of experiments were also done to provide valuable managerial insights for planners. Lastly, the performance of the proposed solution method was compared with a commercial MIP solver for problem instances of increasing size, which were based on real routing plans executed by the company.

5.1. Case study

This study was motivated by a real-life application in the wood-based panel company, firstly presented in [3]. The focal company owns several mills, each producing a specific portfolio of wood-based panels mainly for furniture, construction and decoration. The case study is at one of the mills in Portugal that produces around 1.2 thousand tons of wood-based panels per day, in a make-to-order basis, and assures its delivery to an average of 30 customers distributed over the entire Iberian Peninsula. The average daily consumption of raw materials is 1,750 m^3. The study uses real data regarding the customers' orders in two of the most representative operational days. There are 30 customers to be visited, whose ordered amounts are in average 35.5 ton/customer, varying between 0.05 and 399 ton. The values of the model parameters are an approximation of those provided by the planners. The distances between locations were computed by resorting to the Google Maps routing engine.

Nowadays, the outbound routes are planned to start in the morning of the next day at the opening hour of the mill of ori-

Table 2
Pre-processing the problem network.

Origin \ Destination	{0} depot or mill of origin	Linehaul l	Linehaul $l' \neq l$	Backhaul b	Mill m	{0} *fictitious depot*
{0} depot or mill of origin		•	•	(*)		
Linehaul l			•	(**)		•
Linehaul $l' \neq l$		•		(**)		•
Backhaul b					(+)	
Mill m						•
{0} *fictitious depot*						

Legend:
(*) is generated in IIOP but not in OBP;
(**) for each linehaul, generate the arcs to all the backhauls that lead to a cost-effective solution (i.e. satisfy line 5 of Algorithm 1);
(+) for each backhaul, only generate the arc to the minimum cost mill.

gin. It is assumed that all routes can start at the same time, and there are no time windows conditioning the time of arrival to customers, suppliers or mills. The responsible for outbound logistics determines the exact number of trucks needed for the next day and groups the customers to be visited in each route according to empirical rules that rely on the customers' geographical location. Then, the routes are assigned to the third-party logistics operators (3PL) with whom there are valid outsourced contracts. The generic contractual conditions are a fixed cost of 70€ per truck used and a variable cost of 1.7€ per km travelled. The fleet available at the mill of origin in each day encompassed 100 trucks, of which 20 trucks have capacity up to 10 tons, 40 trucks have a capacity of 20 tons, and 40 trucks have a capacity of 40 tons, summing up a total transportation capacity of 2600 ton. Each vehicle must deliver at least 0.5 ton to each customer visited (i.e., $\psi = 0.5$), except when its demand is lower than this parameter. All trucks are prepared to do IRs, if needed.

Overall, the current logistics process results in a low rate of inbound-outbound flow integration, and the logistics planner has very little visibility about the arrival time to the customers and the time and characteristics of the inbound loads.

The 3PL assigns a truck driver to each route. Then, the driver is responsible for establishing the sequence for visiting all outbound customers, the path and schedules, which may or may not be optimal. The decision of either to visit a supplier (backhaul) or not is often taken by the driver, based on the extra cost for visiting a known supplier in the vicinity of the last costumer (linehaul) visited in the route, in case it is doable within the route maximum duration length set by the 3PL business rules and conditioned by transportation legislation. There are 75 possible suppliers of woodchips for IRs. The raw materials may be delivered back to the mill of origin or alternatively to any of the other three mills owned by the company in the Iberian Peninsula. Currently, there is no minimum amount of backhauling required. According to the experience of the planners, the reward for each backhauling can go up to 10 € /ton.

The computational results were obtained with two major groups of instances (A and B) built upon the previous case study description, each one of them corresponding to a representative operational day. Instances A differ from instances B with respect to the average distance between the linehauls. The linehauls in instances A are more geographically dispersed, with an average distance between linehauls and depot of 461 km, while in instances B it is 197 km. Baseline instances A30 and B30 correspond to the situation described, with 30 linehauls, a total demand of 1853 ton, 75 backhauls, 4 mills and 100 available trucks. Instances among the same instance group differ in the number of linehauls to visit (10, 30 or 50 linehauls) and the number of possible backhauls (0, 25, 50, 75 or 100 backhauls). The instances were generated in a cumulative manner, i.e., the largest instances contain all locations considered in smaller instances. The selection of the locations to be included/removed in the instances was performed randomly from the dataset of the case study.

5.2. Comparison among distinct planning strategies

Instances A10 and B10 were used in these experiments to compare and quantify the benefits of adopting an OBP or IIOP strategy versus DIOP because it is possible to solve the model quickly to optimality while larger instances require the proposed matheuristic whose gaps to optimality could bias the results. Furthermore, the resulting routing plan can be easily visualized.

To perform this comparison, two different reward values were considered (1€ /ton and 7€ /ton). In order to avoid results biased by different reward values for IRs and OIRs, the backhauling reward was set regardless of the type of route (whether it was a dedicated backhaul route or an opportunistic one) and is generically called reward instead of backhauling reward. The inbound quantity to be satisfied was set to 160 ton of raw materials, which corresponds to approximately twice the outbound quantities of finished products in these instances, taking into account the mills' productive efficiency. The remaining parameters remained unchanged throughout the instances, with $\alpha = 1200$ km and $\psi = 0.5$ ton.

For the IIOP strategy, model [P2] was solved to optimality and a given backhauling amount was set. The DIOP models [P3] and [P4] were also solved to optimality with this same backhauling amount to allow a fair comparison. In respect to the OBP strategy, the rationale to allow its comparison with the remaining strategies consisted in: (i) solving the OBP model [P0] to optimality, replacing constraints (15) by a similar set of constraints where a maximum (instead of minimum) backhauling amount of 160 ton is set; (ii) solving model [P4] to obtain the IRs for the differential amount between 160 ton and the already backhauled amount via OBP; (iii) computing the total costs for these two models.

The obtained results are presented in Table 3. In these instances, the matheuristic was not required, since the computational time for proving optimality in the solver was very short (less than 5 min on average). In these experiments, the number of binary decision variables ranged from 10,000 to 17,000.

The analysis of these results shows that the logistics planning strategy leading to the lowest cost is IIOP in all the experiments. In some cases, the strategy OBP performs better than DIOP, as intuitively expected, but in others, it does not. This is because the OBP model is myopic in the sense that it includes all OIRs that are cost-effective for a given backhauling reward value, but does not trade-off between OIRs and IRs as it happens with the IIOP model.

Instance B10 with a reward value of 7€ /ton exemplifies a case where the IIOP strategy is better than OBP and better than DIOP

Table 3
Comparison between alternative Inbound and Outbound Planning strategies.

Reward (€ /ton)	Instance	Planning strategy	Objective Function	Costs (€)			No. routes				Backhauled amount (ton)	No. trucks used	Runtime (s)
				Total	Fixed	Transport	Total	OIR	OR	IR			
7.00	A10	Integrated	2536	3656	350	3306	7	0	3	4	160	5	821
		Opportunistic	2714	3834	420	3414	7	1	3	3	160	6	45
		Decoupled	2536	3656	350	3306	7	0	3	4	160	5	37
	B10	Integrated	768	1888	280	1608	4	3	0	1	160	4	36
		Opportunistic	771	1891	280	1611	4	4	0	0	160	4	32
		Decoupled	896	2016	350	1666	7	0	3	4	160	5	517
1.00	A10	Integrated	3496	3656	350	3306	7	0	3	4	160	5	60
		Opportunistic	3496	3656	350	3306	7	0	3	4	160	5	47
		Decoupled	3496	3656	350	3306	7	0	3	4	160	5	34
	B10	Integrated	1802	1962	350	1612	5	4	1	0	160	5	21
		Opportunistic	1811	1971	350	1621	6	1	2	3	160	5	332
		Decoupled	1856	2016	350	1666	7	0	3	4	160	5	243

strategies. The total costs of the resulting logistics plans are 1,888€, 1,891€, and 2,016€ respectively. The optimal IIOP routing plan consists of three OIRs (for trucks k_1, k_2 and k_5) and one IR (for truck k_4) (Fig. 4a). While, the optimal plan for OBP encompasses three OIRs identical to the later plan, and one extra OIR (k_3) (Fig. 4b). The OIR k_3 is still cost-effective for that reward value, but it is costlier than doing the alternative IR k_4 as in the IIOP plan. No IRs are foreseen in the OBP strategy because the routing plan obtained by solving model [P0] already fulfils the whole demanded backhauled amount; therefore there is no stimulus for finding IRs with model [P4] afterwards.

The decoupled planning strategy for instance B10 leads to a 6.8% increase of the total costs when compared with the previous, due to the increase of the transportation costs and also the use of five vehicles instead of four (Fig. 4c). The overall routing plan encompasses four IRs (obtained with model [P4]), plus three ORs (obtained with model [P3]). The IRs are similar to the ones of vehicle k_4 in the IIOP strategy but the ORs are not. This is because the linehauls are re-distributed in the routes in a different way when the visit to backhauls is not considered in the same model. For example, linehaul 7 was split deliveries according to the IIOP and OBP plans due to its geographical proximity to the backhaul 13. This no longer happens in the decoupled planning, and this linehaul is visited only once in the course of a longer route that extends up to linehaul 10.

Conversely, instance A10 with a reward value of 7€ /ton, exemplifies a case where the performance of the IIOP strategy is the same as the DIOP strategy (3,656€), and the OBP performs worse than the other planning strategies (3,834€, 4.9% worse). The optimal IIOP routing solution consists of three ORs and four IRs (Fig. 4a). In this setting, with the linehauls more geographically dispersed and farther from the suppliers and neighbouring mills, it is cheaper to visit several times supplier 16 in dedicated IRs than considering OIRs. However, the solution of the OBP model, which is myopic with respect to this possibility, encompasses one OIR that visits the cost-effective backhaul 22 (Fig. 4f).

The analysis of these results also shows that the backhauling reward value has a significant impact on the routing plans and can lead to different conclusions with respect to the comparison between the alternative planning strategies. For example, when the reward value is lowered to 1€ /ton, the results for instance B10 show that the visit to backhauls 15 and 16 are no longer cost-effective in the IIOP strategy. Hence, the routing plan consists of four OIRs, all visiting backhaul 13 (Fig. 4d) and 1 OR. The total costs are 3.8% higher than in the experiment with a reward of 7€ /ton, due to an increase in the total transportation distance and in the use of five vehicles instead of four.

It is noteworthy that lowering the value of the reward for visiting the backhauls has a negative effect on revenue and consequently, increasing the value of the objective function (134% higher than with previous experiment with 7€). For this case, the IIOP strategy still performs better than the OBP and DIOP. However, the total cost savings are reduced to 2.7% and 2.2%, respectively. This is due to the fact that with a lower reward value, the use of OIRs is less attractive, and the inbound demand must, therefore, be satisfied with dedicated backhaul routes.

In instance A10, when the reward value is lowered to 1€ per ton, there is no visit to a backhaul that is cost-effective. Hence the optimal plan for the OBP strategy does not consider any OIR, and it is identical to the IIOP and DIOP strategies described before.

These findings suggest that IIOP is the strategy that allows the optimisation of the combination between backhauling and inbound routing, but under specific circumstances that favour the supply of raw materials through cost-effective OIRs instead of direct IRs, OBP can perform better than DIOP. As shown in these experiments, these circumstances depend on the backhauling reward value for visiting a backhaul in an OIR and on the geographical configuration of the logistics network of the planning day, especially the relative distance between a linehaul that can be visited last in a route, and a neighbouring backhaul and mill.

As stated before, the opportunistic planning strategy can be considered an "intermediate" stage from DIOP towards IIOP. The transition from DIOP to opportunistic planning is smoother since it is restricted to organisational changes within the local outbound logistics offices in each mill, while the IIOP also impacts in the central office currently responsible for inbound logistics planning. During this intermediate stage using the OBP strategy, the planners need to compare the optimal routing plan with the outcome of the DIOP strategy in order to establish if backhauling is favourable for the set of customers visited in each day.

5.3. Managerial insights

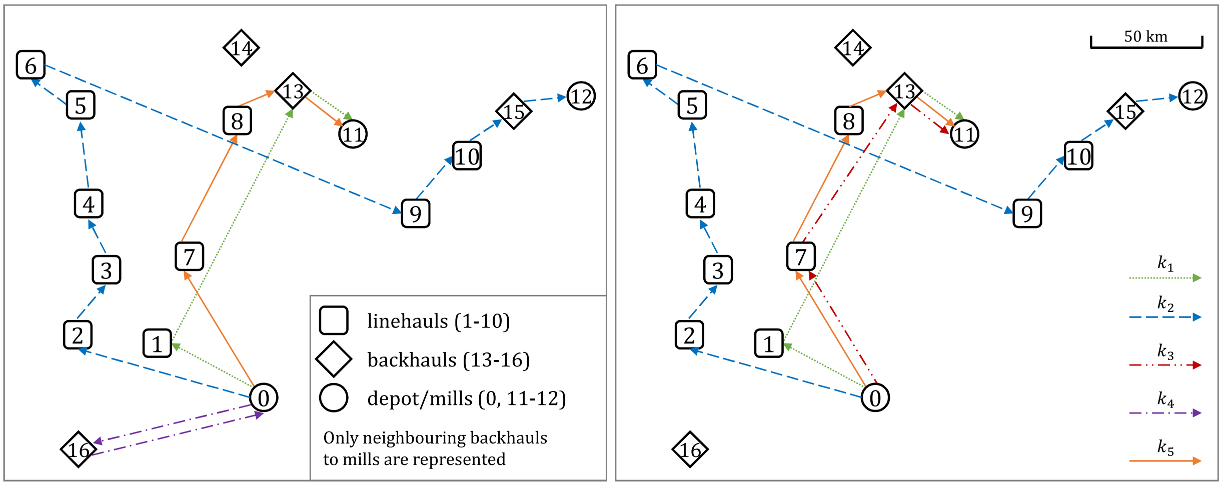
Focusing on OBP, which is the strategy likelier to be adopted by the planners in this case study, additional experiments were designed for instances with 10 linehauls (A10, B10), to provide managerial insights on how the values of key parameters of model [P0] set by the planners with some degree of uncertainty, may actually impact on the routing plan. The parameters under study are:

- backhauling reward (δ), i.e., incentive for picking up one unit of raw material in a backhaul $b \in B$ and delivering it to a mill $m \in M$. For simplification purposes, it is assumed that the reward is the same for all backhauls and mills that accept compatible types of raw materials. Based on experts opinions, the reward can vary in a range from 1 to 10 euros per ton;
- minimum backhauled amount (β), i.e., amount of raw materials to be backhauled in OIR. If parameter $\beta = 0$ then no visits to backhauls are required; if $\beta \geq 1$ ton, then, at least, one backhaul must be included; and if $\beta \geq 41$ ton, then, at least, two backhauls must be included), since the maximum truck capacity is 40 ton;
- maximum length of the route (α), i.e., maximum distance travelled in a route. The values tested are 1,200 km (corresponding to the longest distance from the mill to a linehaul in instance A10) and 1,500 km;
- minimum delivery (ψ), i.e., the minimum allowable amount of order delivered to a linehaul, conditioning the possibility of splitting the order of a linehaul into more than one deliveries done by different trucks. The values tested were $\psi = 0.5$ ton, meaning that many split deliveries are allowed, $\psi = 5$ ton and $\psi = 10$ ton (corresponding to the capacity of the smallest truck), meaning that split deliveries are more restricted.

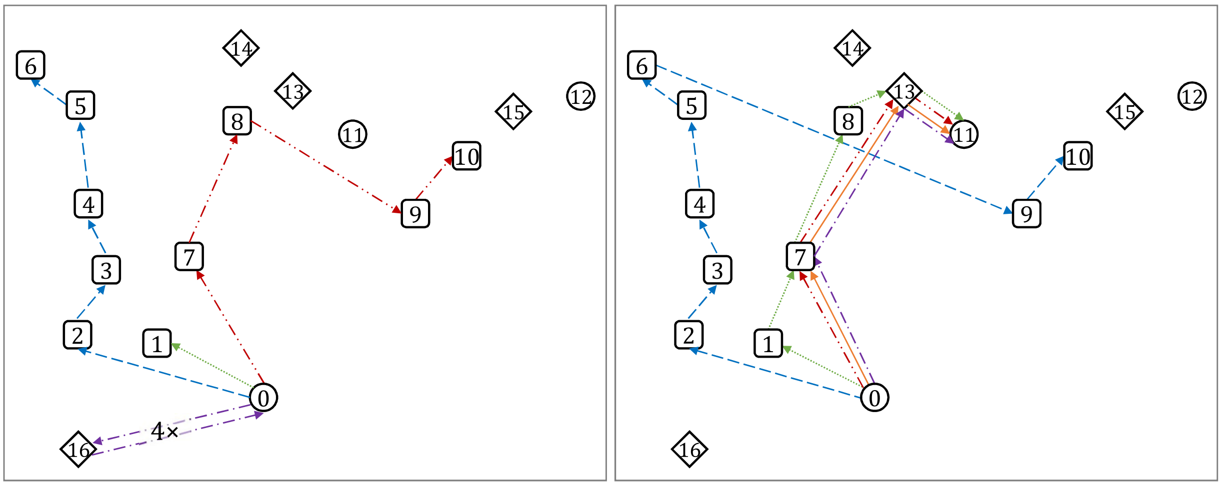
Let us state the baseline conditions for this analysis $\delta = 6$ € /ton for instance A10 and $\delta = 1$ € /ton for instance B10, $\alpha = 1200$ km for A10 and $\alpha = 400$ km for B10, $\beta = 0$ and $\psi = 0.5$ ton. The results of the experiments presented in Table 4 and in the Appendix are the basis for managerial insights that can be valuable for route planning.

Impact of the variation of the value of reward for visiting a backhaul (δ)

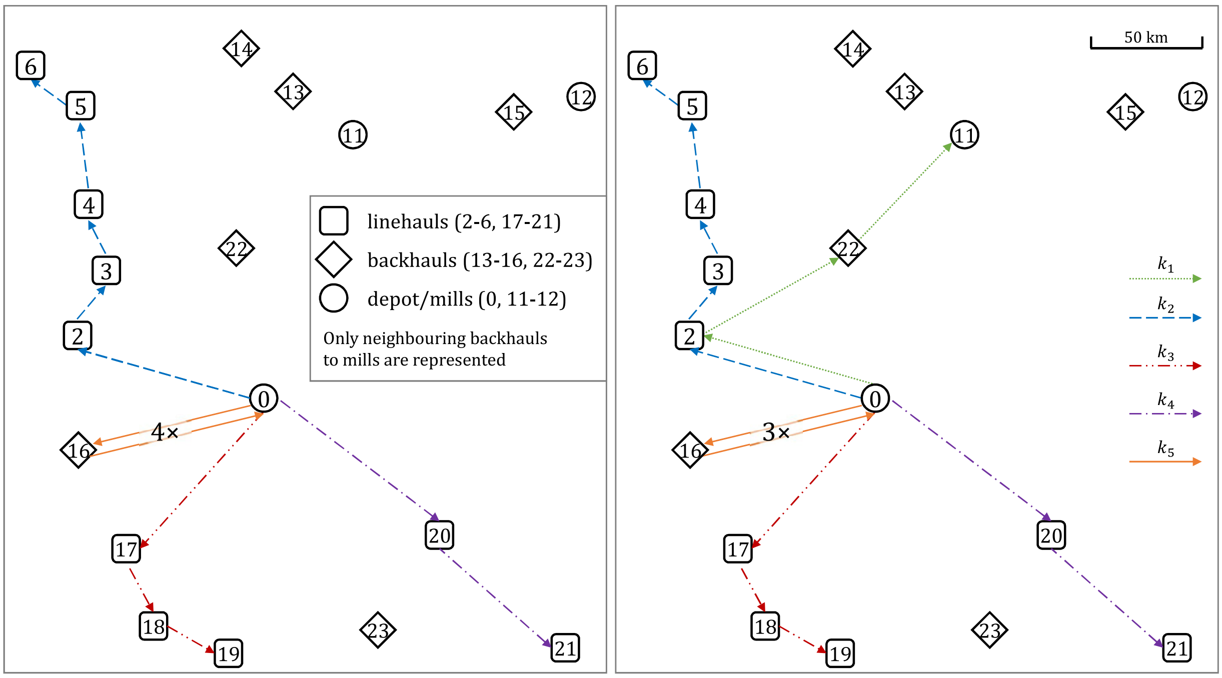
Results generically confirm a positive effect in the objective function of increasing the value of δ, because more OIRs are performed, often with the same number of trucks. The total transportation costs increase, due to the increase in the total distances travelled, but these are compensated with a higher total reward collected. The first managerial insight that can be formulated is that planners wishing to foster an increase of OIRs should set the reward value at least equal to the extra travelling costs for visiting the most cost-effective backhaul (i.e. the costs for travelling



(a) B10, integrated strategy (7€/ton) (b) B10, opportunistic strategy (7€/ton)



(c) B10, decoupled strategies (1 and 7€/ton) (d) B10, integrated strategy (1€/ton)



(e) A10, integrated, decoupled (1 and 7€/ton) and opportunistic strategies (1€/ton) (f) A10, opportunistic (7€/ton)

Fig. 4. Graphical representation of the planning strategies for instances A10 and B10.

Table 4
Summary of the experiments on the impact of the values of model parameters.

Inst.	Parameter				Objective Function	Costs (€)			Routes			Runtime (s)	MIP Gap
	α	β	ψ	δ		Total	Fixed	Transport	Total	OIR	OR		
A10	**1,200**	**0**	**0.5**	**6**	**4,703**	**4,703**	**280**	**4,423**	**4**	**0**	**4**	**19**	**1.8%**
	–	–	–	7	−1%	+5%	+25%	+4%	+1	+1	0	105	0.6%
	–	–	–	8	−2%	+5%	+25%	+4%	+1	+1	0	112	0.3%
	–	–	–	10	−46%	+243%	+800%	+208%	+32	+34	−2	272	1.9%
	–	1	–	–	0%	+5%	+25%	+4%	+1	+1	0	19	1.0%
	–	41	–	–	+2%	+12%	+50%	+10%	+2	+2	0	190	0.0%
	1500	–	–	–	−32%	−32%	−25%	−32%	−1	0	−1	26	1.6%
	–	–	5	–	+1%	+6%	+25%	+4%	+1	+1	0	28	2.0%
	–	–	5	10	−7%	+35%	+75%	+33%	+3	+5	−2	304	0.4%
	–	–	10	–	+1%	+6%	+25%	+4%	+1	+1	0	130	1.9%
	–	–	10	10	−5%	+21%	+25%	+21%	+1	+3	−2	308	0.7%
B10	**400**	**0**	**0.5**	**1**	**2,002**	**2,002**	**210**	**1,792**	**3**	**3**	**0**	**16**	**0.0%**
	–	–	–	2	0%	+4%	+33%	+1%	+1	−2	+3	28	0.0%
	–	–	–	5	−105%	+284%	+1,300%	+165%	+39	+36	+3	32	0.0%
	–	1	–	–	+2%	+4%	+33%	+1%	+1	−2	+3	39	0.0%
	–	41	–	–	+7%	+11%	+67%	+4%	+2	−1	+3	41	0.0%
	800	–	–	–	−22%	−20%	0%	−22%	0	−2	+2	72	0.0%
	–	–	5	–	0%	0%	0%	0%	0	−3	+3	25	0.0%
	–	–	5	5	−23%	+57%	+267%	+32%	+8	+5	+3	20	0.0%
	–	–	10	–	0%	0%	0%	0%	0	−3	+3	29	0.0%
	–	–	10	5	−15%	+25%	+133%	+13%	+4	+1	+3	18	0.0%

Legend: α: maximum length of the route (km); β: minimum backhauled amount (ton); ψ: minimum delivery amount (ton); δ: backhauling reward (€ /ton); Runtime: computational time after which no better solution was obtained (seconds); MIP Gap: percentual difference obtained by Gurobi between the upper and lower bound of the branch-and-bound method. All models were run with a maximum time limit of 3,600s. The first row of results for each instance (highlighted in bold) contains the baseline values, and the rows that follow exhibit either the absolute or the percentual variation compared with the baseline values (except for Runtime and MIP Gap values).

from the last linehaul to the backhaul and from there to the closest mill).

For instance A10 the minimum δ should be 7€ /ton. Below that value, there is no backhaul that is cost-effective, hence, no OIRs are included in the optimal routing plan. The number of trucks needed increases for four to five. Increasing δ to 8 € /ton improve the value of the objective function but do not change the costs, because the number of trucks and the routing plan remains the same. However, very high values of δ are not beneficial as it leads to the use of a large truck fleet. Hence, the percentual increase of total costs is much higher than the gains in the value of the objective function, and the resulting routing plan is hardly adopted in practice. For example, a δ equal to 10€ /ton in instance A10 leads to costs 243% higher than in the baseline, corresponding to the highest number of 34 OIRs out of the 36 routes that compose the optimal routing plan.

Regarding instance B10, the linehauls are less geographically dispersed than in A10; thus, a slight increase in the δ leads to significant changes in the number of OIRs and the improvement in the objective function value. In fact, the baseline experiment with a δ equal to 1€ /ton already leads to 3 OIRs and one for each of the trucks used. For δ equal or higher than 5€ /ton the routing plan changes drastically to 39 OIRs requiring 39 extra trucks.

Impact of the variation of the required backhauled amount (β)

Experiments suggest that increasing β has a negative impact on the value of the objective function because it increases the transportation costs for the mandatory visit to backhaul. However, in some instances, such as A10, it leads to an increase in OIRs, while in others, such as B10, it leads to an increase of the number of IRs. A second managerial insight for planners relates to the fact that the geographical dispersion between the linehauls, backhauls and mills is the determining factor for finding the optimal routing plan, as discussed in Section 5.2. It is also noteworthy that, under some circumstances (e.g. for $\delta \leq 2$ and $\beta > 0$ for A10), the solution turns infeasible because the pre-processing algorithm guarantees that only cost-efficient integrated routes can be created.

Impact of the variation of the delivery amount at a linehaul (ψ)

Experiments indicate that increasing ψ has a slightly negative effect on the value of the objective function. Although this may imply the use of fewer vehicles, this also decreases the possibility of creating integrated routes and, as such, the possibility of collecting a higher total reward.

There is a complementary relation between the key parameters ψ and δ in fostering the number of OIRs in the optimal routing plan. In practice, if ψ is low, means that more visits to the linehauls are allowed, and so, there is more flexibility in the routing plan to include OIR, especially if the reward for visiting a backhaul is high. In fact, the number of OIRs is maximized (34 out of 35 routes) if ψ is very low (e.g., 0.5 ton) and δ is very high (e.g., 10€). However, these high number of integrated routes (e.g. 34 out of 36 routes) can hardly represent the common practice (Fig. 5). Hence, another managerial insight for planners relates to the importance of properly addressing the trade-off between the offered reward and the maximum number of visits allowed to a linehaul, which is specific for each case.

Impact of the variation of the maximum length of the route (α)

Experiments show that increasing α tends to improve the objective function, due to the decrease in the number of required vehicles and the possibility to visit a larger number of linehauls is the same route. However, without a direct impact on the number of OIR. As an example for instance A10, increasing α from 1,200 km to 1,500 km, all other parameters remaining the same as in the baseline scenario, lead to a decrease of 32% in the value of the objective function, related with the use of 3 vehicles instead of 4. In instance B10, the increase from 400 km to 800 km, leads to a decrease of 22% in the value of the objective function due to longer routes, using the same fleet of 3 trucks.

In summary, results show that there are several trade-offs that need to be analysed by planners to balance the increase of OIRs and the increase in transportation costs. In particular, results suggest that α is the parameter that impacts the most in improving the value of the objective function and costs (improvements of 32%

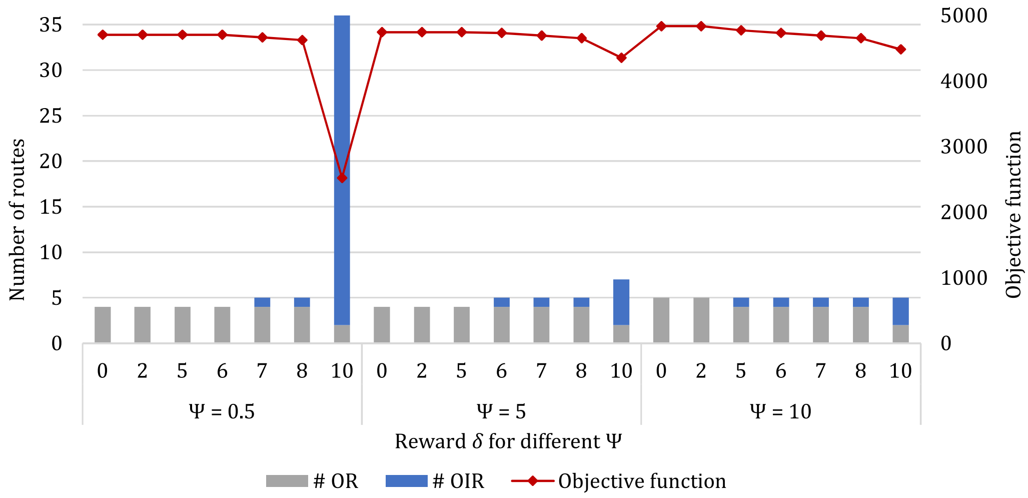


Fig. 5. Impact of the variation of the reward (δ) and minimum delivery (ψ) ($\alpha = 1,200$, $\beta = 0$).

in instance A10 e 22% in B10, because it enables to use fewer vehicles, and fewer distances travelled, however, do not necessarily foster OIRs.

Moreover, the main parameter to be taken into account for planners willing to improve OIRs is δ. As discussed before, OIRs tend to be included in the routing plan when the reward value is above a threshold, corresponding to visiting the first cost-effective backhaul. The value of this threshold depends on the geographical dispersion of nodes in the transportation network and particularly the distance between the last visited linehaul, the closest backhaul and its neighbouring mill.

5.4. Performance of the solution approach

Despite the fact that the solver is able to obtain optimal solutions within a few minutes for problem instances of 10 linehauls, this is not the case for larger instances. In these cases, the use of the matheuristic is justified in order to obtain good quality solutions in a shorter computational time. A set of computational experiments was envisaged to validate the proposed solution approach. Instances of group A and B were solved using the standalone MIP solver approach and the fix-and-optimise matheuristic. These experiments were performed in an Intel Xeon E5-2450 @ 2.10GHz CPU with capacity for 16 simultaneous processing threads.

Both approaches were run for 3,600s, with $\alpha = 1,200$, $\beta = 0$ and $\psi = 0.5$. The MIP solver was executed once for each instance using Gurobi's default parameters and the fix-and-optimise approach was run 10 times for each instance, using the parameters described in Table 5.

Table 6 summarizes the computational results of the MIP solver and matheuristic approaches for the 30 problem instances.

The results demonstrate that both the MIP solver and the matheuristic are adequate for solving instances up to 10 linehauls (groups A10 and B10), as the solver is able to prove optimality for most instances and the matheuristic easily reaches the same solution as the MIP solver. For larger instances, the MIP solver yields optimality gaps up to 32% for instances of groups A30, A50, B30 and B50. Specifically to instances of group A, it is possible to observe the increase in the number of routes that perform backhauling as the number of backhaul locations progressively increases. In instances of group B30, a single OIR is used when backhauling is allowed, and in group B50 no opportunistic backhauling is performed. However, the obtained solutions by the MIP solver when the number of backhauls increases do not necessarily improve, contrarily to what would be expected. Furthermore, for instances of group B50, the solver is unable to find a single feasible solution within the 1-hour limit for 4 out of the 5 instances. This fact is probably due to the increase in model size and complexity when more backhaul locations are being considered, thus requiring more time for Gurobi to reach identical solutions when exploring the branch-and-bound tree.

The matheuristic approach exhibits small standard deviation values for the 10 repetitions performed for each instance, thus suggesting that the obtained results are robust. The negative percentual difference values between the solver and the matheuristic suggest that the matheuristic is able to converge correctly to better solutions, as opposed to the solver, which exhibits very high optimality gaps. This negative percentual difference tends to be increasingly more expressive with the increase in instance size. Furthermore, results also suggest that the matheuristic also takes better advantage of the increase in the number of backhaul locations,

Table 5
Used parameters for the matheuristic approach.

Parameters		Value
Termination criteria	Time limit	3,600s
No-improvement criterion	Improvement between consecutive iterations lower than	100
`RouteRelease`	Subproblem sizes	4, 6, 8, 16 routes
sub-problem	No-improvement limit to change subproblem size	2 iterations
	MIP solver iteration time limit	Multiples of 15s (according to sub-problem size)
`LocationRelease`	Subproblem sizes	2, 4, 6, 8 linehauls
sub-problem	No-improvement limit to change subproblem size	2 iterations
	MIP solver iteration time limit	Multiples of 15s (according to sub-problem size)

Table 6
Computational results of the MIP and matheuristic approaches for 30 problem instances.

Problem instance						MIP Solver					Fix-and-optimise					
Group	$\|L\|$	$\|B\|$	$\|K\|$	Σq_i	ΣQ_k	OF	MIP Gap	Runtime	No. routes	No. OIRs	Objective Function		Runtime	No. routes	No. OIRs	% diff.
											Average	Standard Deviation				
A10	10	0	100	82	2600	3215	0.0%	50	3	0	3215	0	31	3	0	0.0%
	10	25	100	82	2600	3215	0.0%	205	3	0	3215	0	37	3	0	0.0%
	10	50	100	82	2600	3215	0.0%	60	3	0	3215	0	28	3	0	0.0%
	10	75	100	82	2600	3215	0.0%	55	3	0	3215	0	30	3	0	0.0%
	10	100	100	82	2600	3215	0.0%	31	3	0	3215	0	33	3	0	0.0%
A30	30	0	100	1853	2600	13,745	18.4%	3110	53	0	13,627	19	2957	53	0	−0.9%
	30	25	100	1853	2600	13,696	19.6%	3444	54	26	13,404	20	2885	53	25	−2.1%
	30	50	100	1853	2600	13,701	20.1%	2758	54	32	13,374	21	2879	53	31	−2.4%
	30	75	100	1853	2600	13,833	20.9%	3187	55	33	13,366	19	2576	53	31	−3.4%
	30	100	100	1853	2600	13,702	20.1%	1877	56	34	13,369	20	2731	53	31	−2.4%
A50	50	0	100	2061	2600	20,986	29.9%	2968	65	0	19,465	363	3463	64	0	−7.2%
	50	25	100	2061	2600	19,717	26.5%	1902	64	30	19,019	194	3527	63	26	−3.5%
	50	50	100	2061	2600	19,907	27.3%	3469	65	37	19,079	313	3397	64	31	−4.2%
	50	75	100	2061	2600	20,829	30.5%	3019	65	36	19,167	423	3464	64	32	−8.0%
	50	100	100	2061	2600	21,365	32.3%	2653	66	33	19,116	327	3513	65	36	−10.5%
B10	10	0	100	82	2600	1575	1.9%	129	3	0	1575	0	39	3	0	0.0%
	10	25	100	82	2600	1575	2.7%	117	3	0	1575	0	39	3	0	0.0%
	10	50	100	82	2600	1575	0.0%	53	3	0	1575	0	39	3	0	0.0%
	10	75	100	82	2600	1575	0.0%	51	3	0	1575	0	44	3	0	0.0%
	10	100	100	82	2600	1575	0.0%	51	3	0	1575	0	61	3	0	0.0%
B30	30	0	100	818	2600	8695	20.1%	3032	23	0	8210	9	2267	22	0	−5.6%
	30	25	100	818	2600	8626	19.4%	3354	23	1	8212	15	2730	21	1	−4.8%
	30	50	100	818	2600	9162	30.5%	2473	23	0	8206	12	2292	22	1	−10.4%
	30	75	100	818	2600	8350	16.8%	3262	21	1	8206	10	2876	21	1	−1.7%
	30	100	100	818	2600	9003	32.1%	441	23	1	8198	2	2474	22	1	−8.9%
B50	50	0	100	2054	2600	–	–	–	–	–	45,393	956	3419	68	0	–
	50	25	100	2054	2600	–	–	–	–	–	45,325	793	3420	69	0	–
	50	50	100	2054	2600	–	–	–	–	–	45,759	526	3388	64	0	–
	50	75	100	2054	2600	46,991	13.8%	2830	69	0	45,592	655	3359	65	0	−3.0%
	50	100	100	2054	2600	–	–	–	–	–	45,768	646	3412	67	0	–
Average																**−3.0%**

Legend: $|L|$: number of linehauls to be visited; $|B|$: number of possible backhauls; $|K|$ total number of vehicles available; Σq_i: total quantity to be delivered to linehauls (ton); ΣQ_k: total vehicle transportation capacity (ton); OF: Final value of the objective function (€); Runtime: average computational time after which no better solution was obtained (seconds); MIP Gap: Percentual difference obtained by Gurobi between the upper and lower bounds of the branch-and-bound method; No. routes: total number of routes; No. OIR: number of routes that include visit to a backhaul; % diff.: percentual difference of the fix-and-optimise average objective function towards Gurobi's objective function (a negative difference favours the matheuristic). For the fix-and-optimise approach, the route indicators correspond to the repetition whose objective function value was closest to the obtained average.

as the objective function values generally decrease when the number of backhaul locations increases.

From these results, we can say that the proposed matheuristic approach is adequate for solving the problem at hand. For instances of considerable size, the MIP solver starts to struggle in finding feasible solutions in an acceptable time limit, and apparently also has greater difficulties taking advantage of backhauling, while the matheuristic is able to decrease the overall logistics costs with an increase in the number of backhaul locations, therefore yielding more consistent results.

6. Conclusions and future work

Integrating planning processes requires a thorough assessment of both quantitative benefits pertaining to the expected decrease in the related costs and qualitative impacts related to the usual need of breaking functional silos. This work explored, mainly, the quantitative aspect of integrating outbound and inbound logistics routes. We used as a background a case-study from the wood-based panel industry, but the results and approaches developed are generalisable for other settings in which this integration can be modelled as an rVRPB (e.g., grocery retail, cement distribution). Besides modelling three possible planning strategies (i.e., OBP, IIOP, and DIOP), we have also developed a matheuristic to tackle real-world instances of this problem.

Three key conclusions emerge from our computational study. Firstly, the intuitive idea that intermediate levels of integration would always result in better planning outcomes was not verified. DIOP outperforms OBP in certain geographical contexts where the distribution network is more dispersed. In our studies, this happened in instance A10 when the average distance between linehaul customers and the depot of origin is 197 km. In this case, it was actually cheaper to assure the supply of raw material through dedicated inbound routes (i.e. going to and from the nearest supplier) than including a visit to a supplier at the end of the outbound route (i.e. after visiting all customers). The IIOP model does this trade-off, but the OBP model is myopic to the possibility of doing direct inbound routes, hence, leading to worse results than DIOP.

Secondly, we confirm that there are important parameters dealt by the planners with some degree of uncertainty that actually can have a great influence on the total costs of the routing plan. This study analysed four of these parameters - backhauling reward, minimum backhauled amount, maximum length of the route and minimum delivery amount allowed. Results suggest that increasing the maximum length of the route leads to the largest impact in the performance of the routing plan but including a quantitative reward for each supplier visited will likelier increase the proportion of integrated inbound and outbound routes in the overall routing plan. In fact, the total reward (€ /ton) should be equal or higher than the extra transportation costs for the most cost-effective supplier. Meaning that the extra distance travelled empty from the last customer to the nearest supplier and then full from there to the neighbouring mill is minimized.

Finally, the developed matheuristic proved to be a suitable approach to tackle this problem and this fact reiterated the interest of fix-and-optimise to solve routing problems.

Future work could be devoted to merging the qualitative and quantitative assessments related to the integration of planning processes. In particular, the study of integrated inbound and outbound routes is of interest due to its potential in improving the ever-relevant sustainability dimension.

Acknowledgements

This work is financed by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e Tecnologia within project POCI-01-0145-FEDER-016733.

Further funding was received from the Portugal 2020 project “DM4Manufacturing - Aligning manufacturing decision making with advanced manufacturing technologies”, POCI-01-0145-FEDER-016418, financed by UE/FEDER through the program COMPETE2020.

The second author also acknowledges the support of the Ph.D. Grant SFRH/BD/136314/2018, awarded by the Portuguese Science and Technology Foundation (FCT) and financed by the European Social Fund (ESF) and by National Funds of the Portuguese Ministry of Science, Technology and Higher Education (MCTES) through the Human Capital Operational Programme (POCH).

We would also like to thank the wood-based products company for the cooperation and for supplying all the data.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.omega.2019.102172.

References

[1] Adulyasak Y, Cordeau J-F, Jans R. The production routing problem: a review of formulations and solution algorithms. Comput Oper Res 2015;55:141–52. doi:10.1016/j.cor.2014.01.011.
[2] Amorim P, Günther H-O, Almada-Lobo B. Multi-objective integrated production and distribution planning of perishable products. Int J Prod Econ 2012;138(1):89–101. doi:10.1016/j.ijpe.2012.03.005.
[3] Amorim P, Marques A, Oliveira B. Wood-based products distribution planning with wood returns in a iberian company. MOSIM 2014, 10ème Conférence Francophone de Modélisation, Optimisation et Simulation. Nancy, France; 2014.
[4] Aras N, Aksen D, Tekin MT. Selective multi-depot vehicle routing problem with pricing. Transp Res Part C 2011;19(5):866–84. doi:10.1016/j.trc.2010.08.003.
[5] Archetti C, Speranza MG. The inventory routing problem: the value of integration. Int Trans Oper Res 2015;23(3):393–407. doi:10.1111/itor.12226.
[6] Audy J-F, Lehoux N, D'Amours S, Rönnqvist M. A framework for an efficient implementation of logistics collaborations. Int Trans Oper Res 2010;19(5):633–57. doi:10.1111/j.1475-3995.2010.00799.x.
[7] Bektaş T, Laporte G, Vigo D. Integrated vehicle routing problems. Comput Oper Res 2015;55:126. doi:10.1016/j.cor.2014.08.008.
[8] Bouchard M, D'Amours S, Rönnqvist M, Azouzi R, Gunn E. Integrated optimization of strategic and tactical planning decisions in forestry. Eur J Oper Res 2017;259(3):1132–43. doi:10.1016/j.ejor.2016.11.022.
[9] Caceres-Cruz J, Arias P, Guimarans D, Riera D, Juan AA. Rich vehicle routing problem. ACM Comput Surv 2014;47(2):1–28. doi:10.1145/2666003.
[10] Carlsson D, Rönnqvist M. Backhauling in forest transportation: models, methods, and practical usage. Can J Forest Res 2007;37(12):2612–23. doi:10.1139/x07-106.
[11] Chávez JJS, Escobar JW, Echeverri MG, Meneses CAP. A metaheuristic ACO to solve the multi-depot vehicle routing problem with backhauls. Int J Ind Eng-Manage 2015;6(2):49–58.
[12] Darvish M, Archetti C, Coelho LC, Speranza MG. Flexible two-echelon location routing problem. Eur J Oper Res 2019;277(3):1124–36. doi:10.1016/j.ejor.2019.04.002.
[13] Davis LB, Sengul I, Ivy JS, Brock LG, Miles L. Scheduling food bank collections and deliveries to ensure food safety and improve access. Socio-Econ Plann Sci 2014;48(3):175–88. doi:10.1016/j.seps.2014.04.001.
[14] Deif I, Bodin LD. Extension of the Clarke and Wright algorithm for solving the vehicle routing problem with backhauling. In: Proceedings of the Babson conference on software uses in transportation and logistic management. Babson Parl; 1984. p. 75–96.
[15] Derigs U, Pullmann M, Vogel U, Oberscheider M, Gronalt M, Hirsch P. Multilevel neighborhood search for solving full truckload routing problems arising in timber transportation. Electron Notes Discrete Mathe 2012;39:281–8. doi:10.1016/j.endm.2012.10.037.
[16] Dominguez O, Guimarans D, Juan AA, de la Nuez I. A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls. Eur J Oper Res 2016;255(2):442–62. doi:10.1016/j.ejor.2016.05.002.
[17] Eguia I, Racero J, Molina JC, Guerrero F. Environmental issues in vehicle routing problems. In: Sustainability appraisal: quantitative methods and mathematical techniques for environmental performance evaluation. Springer Berlin Heidelberg; 2013. p. 215–41. doi:10.1007/978-3-642-32081-1_10.
[18] Goetschalckx M, Jacobs-Blecha C. The vehicle routing problem with backhauls. Eur J Oper Res 1989;42(1):39–51. doi:10.1016/0377-2217(89)90057-x.
[19] Gribkovskaia I, Laporte G, Shyshou A. The single vehicle routing problem with deliveries and selective pickups. Comput Oper Res 2008;35(9):2908–24. doi:10.1016/j.cor.2007.01.007.
[20] Gutiérrez-Jarpa G, Desaulniers G, Laporte G, Marianov V. A branch-and-price algorithm for the vehicle routing problem with deliveries, selective pickups and time windows. Eur J Oper Res 2010;206(2):341–9. doi:10.1016/j.ejor.2010.02.037.
[21] Gutiérrez-Jarpa G, Marianov V, Obreque C. A single vehicle routing problem with fixed delivery and optional collections. IIE Trans 2009;41(12):1067–79. doi:10.1080/07408170903113771.
[22] Hansen P, Mladenović N, Perez-Britos D. Variable neighborhood decomposition search. J Heuristics 2001;7(4):335–50. doi:10.1023/a:1011336210885.
[23] Hirsch P. Minimizing empty truck loads in round timber transport with tabu search strategies. Int J Inf SystSupply Chain Manage 2011;4(2):15–41. doi:10.4018/jisscm.2011040102.
[24] Koç Ç, Laporte G. Vehicle routing with backhauls: review and research perspectives. Comput Oper Res 2018;91:79–91. doi:10.1016/j.cor.2017.11.003.
[25] Küçükoğlu İ, Öztürk N. A differential evolution approach for the vehicle routing problem with backhauls and time windows. J Adv Transp 2013;48(8):942–56. doi:10.1002/atr.1237.
[26] Lahyani R, Khemakhem M, Semet F. Rich vehicle routing problems: from a taxonomy to a definition. Eur J Oper Res 2015;241(1):1–14. doi:10.1016/j.ejor.2014.07.048.
[27] Lai M, Battarra M, Francesco MD, Zuddas P. An adaptive guidance metaheuristic for the vehicle routing problem with splits and clustered backhauls. J Oper Res Soc 2015;66(7):1222–35. doi:10.1057/jors.2014.123.
[28] Lai M, Crainic TG, Francesco MD, Zuddas P. An heuristic search for the routing of heterogeneous trucks with single and double container loads. Transp Res Part E 2013;56:108–18. doi:10.1016/j.tre.2013.06.001.
[29] Larrain H, Coelho LC, Cataldo A. A variable MIP neighborhood descent algorithm for managing inventory and distribution of cash in automated teller machines. Comput Oper Res 2017;85:22–31. doi:10.1016/j.cor.2017.03.010.
[30] Marques AS, Audy JF, D'Amours S, Rónnqvist M. Tactical and operational harvest planning. In: The management of industrial forest plantations. Springer Netherlands; 2014. p. 239–67. doi:10.1007/978-94-017-8899-1_7.
[31] Neves-Moreira F, Almada-Lobo B, Cordeau J-F, Guimarães L, Jans R. Solving a large multi-product production-routing problem with delivery time windows. Omega 2019;86:154–72. doi:10.1016/j.omega.2018.07.006.
[32] Neves-Moreira F, Amorim P, Guimarães L, Almada-Lobo B. A long-haul freight transportation problem: synchronizing resources to deliver requests passing through multiple transshipment locations. Eur J Oper Res 2016;248(2):487–506. doi:10.1016/j.ejor.2015.07.046.
[33] Nguyen PK, Crainic TG, Toulouse M. Multi-trip pickup and delivery problem with time windows and synchronization. Ann Oper Res 2016;253(2):899–934. doi:10.1007/s10479-015-2001-7.
[34] Nikolakopoulos A. A metaheuristic reconstruction algorithm for solving bi-level vehicle routing problems with backhauls for army rapid fielding. In: Operations research/computer science interfaces series. Springer International Publishing; 2014. p. 141–57. doi:10.1007/978-3-319-12075-1_8.
[35] Oesterle J, Bauernhansl T. Exact method for the vehicle routing problem with mixed linehaul and backhaul customers, heterogeneous fleet, time window and manufacturing capacity. Procedia CIRP 2016;41:573–8. doi:10.1016/j.procir.2015.12.040.
[36] Paraphantakul C, Miller-Hooks E, Opasanon S. Scheduling deliveries with backhauls in thailand's cement industry. Transp Res Rec 2012;2269(1):73–82. doi:10.3141/2269-09.
[37] Parragh SN, Doerner KF, Hartl RF. A survey on pickup and delivery problems. J Für Betriebswirtschaft 2008;58(1):21–51. doi:10.1007/s11301-008-0033-7.
[38] Rieck J, Ehrenberg C, Zimmermann J. Many-to-many location-routing with inter-hub transport and multi-commodity pickup-and-delivery. Eur J Oper Res 2014;236(3):863–78. doi:10.1016/j.ejor.2013.12.021.
[39] Ropke S, Pisinger D. A unified heuristic for a large class of vehicle routing problems with backhauls. Eur J Oper Res 2006;171(3):750–75. doi:10.1016/j.ejor.2004.09.004.
[40] Sahling F, Buschkühl L, Tempelmeier H, Helber S. Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-

optimize heuristic. Comput Oper Res 2009;36(9):2546–53. doi:10.1016/j.cor.2008.10.009.

[41] Salhi S, Wassan N, Hajarat M. The fleet size and mix vehicle routing problem with backhauls: formulation and set partitioning-based heuristics. Transp Res Part E 2013;56:22–35. doi:10.1016/j.tre.2013.05.005.

[42] Soares R, Marques A, Amorim P, Rasinmäki J. Multiple vehicle synchronisation in a full truck-load pickup and delivery problem: a case-study in the biomass supply chain. Eur J Oper Res 2019;277(1):174–94. doi:10.1016/j.ejor.2019.02.025.

[43] Speranza MG. Trends in transportation and logistics. Eur J Oper Res 2018;264(3):830–6. doi:10.1016/j.ejor.2016.08.032.

[44] Toth P, Vigo D. Vehicle routing: problems, methods, and applications. Second Edition. SIAM - Society for Industrial and Applied Mathematics; 2014.

[45] Wade AC, Salhi S. An investigation into a new class of vehicle routing problem with backhauls. Omega 2002;30(6):479–87. doi:10.1016/s0305-0483(02)00056-7.

[46] Wassan N, Wassan N, Nagy G, Salhi S. The multiple trip vehicle routing problem with backhauls: formulation and a two-level variable neighbourhood search. Comput Oper Res 2017;78:454–67. doi:10.1016/j.cor.2015.12.017.

[47] Yano CA, Chan TJ, Richter LK, Cutler T, Murty KG, McGettigan D. Vehicle routing at quality stores. Interfaces 1987;17(2):52–63. doi:10.1287/inte.17.2.52.

[48] Yazgitutuncu G, Carreto C, Baker B. A visual interactive approach to classical and mixed vehicle routing problems with backhauls. Omega 2009;37(1):138–54. doi:10.1016/j.omega.2006.11.001.

[49] Yu M, Qi X. A vehicle routing problem with multiple overlapped batches. Transp Res Part E 2014;61:40–55. doi:10.1016/j.tre.2013.10.004.